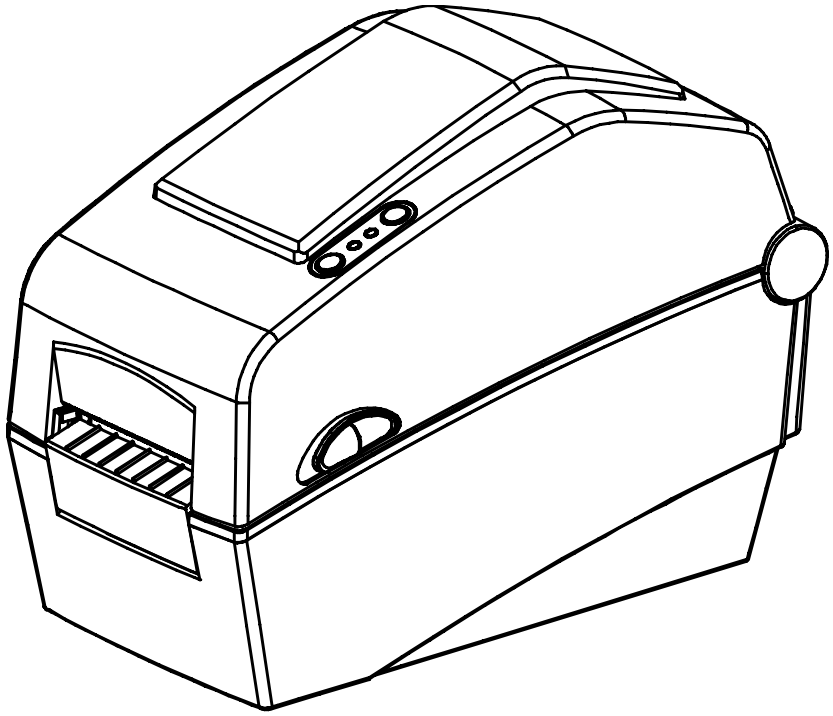
BIXOLON®

Programming Manual

SLCS

Rev. 1.01

SLP-D220 / D220E
SLP-D223 / D223E



http://www.bixolon.com

■ Table of Contents

1. Foreword

In this chapter, the basic concept of SLCS and some information necessary for the programmer to use SLCS will be explained. Please read this part before starting programming for efficient and easy use of BIXOLON Label Printers..

We at BIXOLON maintain ongoing efforts to enhance and upgrade the functions and quality of all our products. In following, product specifications and/or user manual content may be changed without prior notice.

1-1 Image Buffer Configuration

1) Maximum size

A) When using Double Buffering Function

432dots × 1216dots (54mm × 152mm) = 2 inch × 6 inch

B) When using Single Buffering Function

432dots × 2432dots (54mm × 304mm) = 2 inch × 12 inch

2) Dot size : 0.125mm(W) × 0.125mm(H) (203dpi)

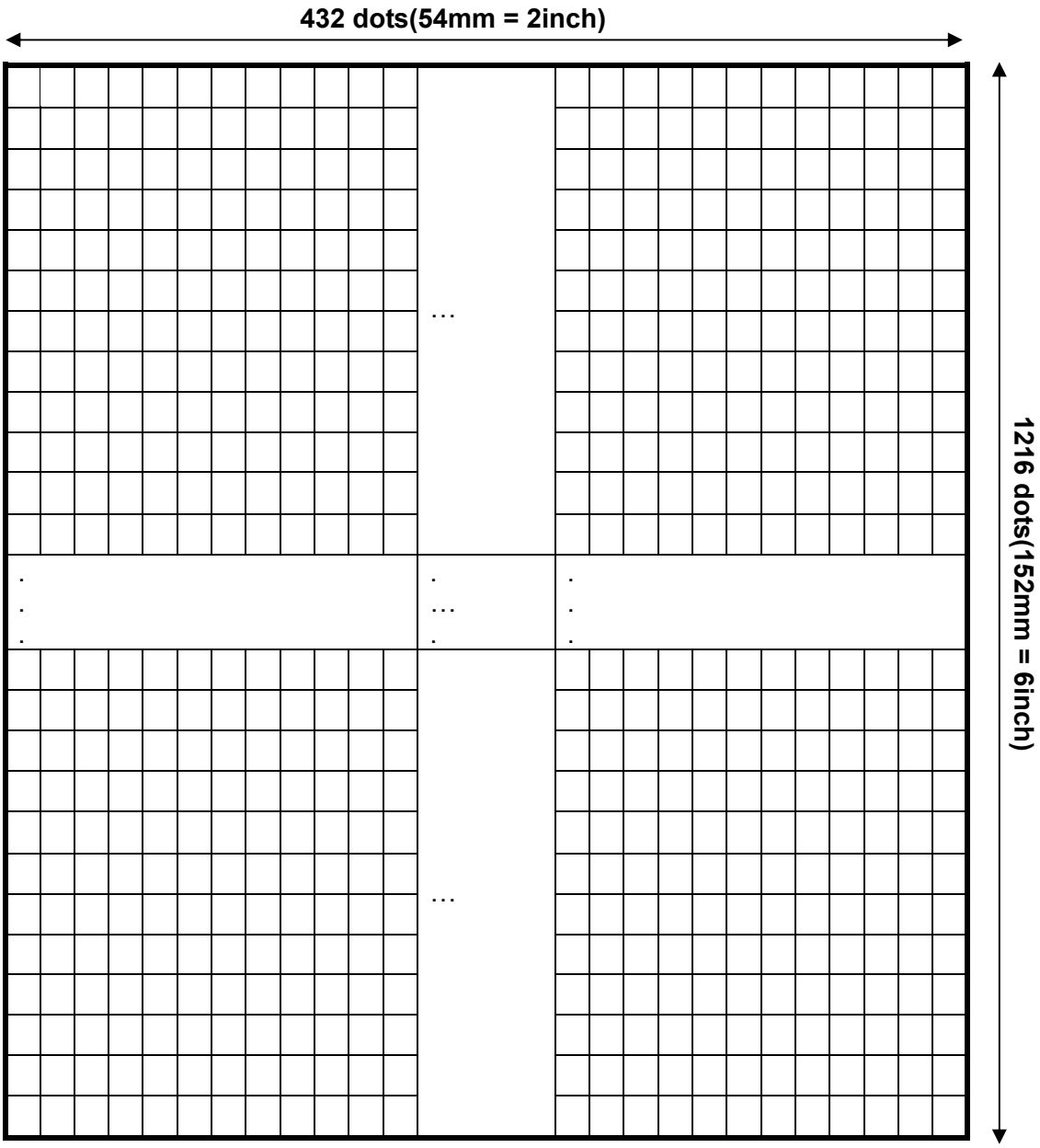


Image Buffer

1-2 Information for calculating position on image buffer

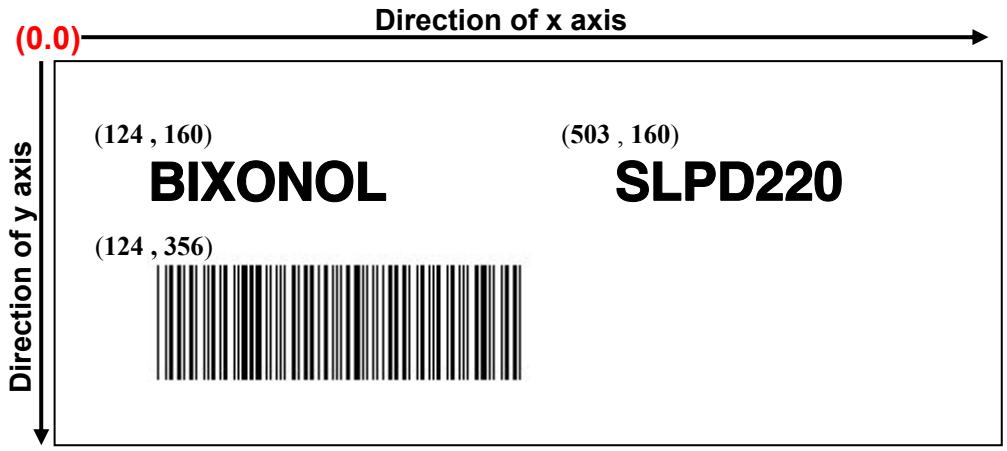
1) Relation between position and number of dots

Inch	mm	dots	Resolution
0.04	1	8	203 dpi
0.40	10.00	80	
1.00	25.40	203	
1.25	31.75	254	
1.50	38.10	305	
1.75	44.45	355	
2.00	50.80	406	
2.25	57.15	457	
2.50	63.50	508	
2.75	69.85	556	
3.00	76.20	610	
4.00	101.6	813	

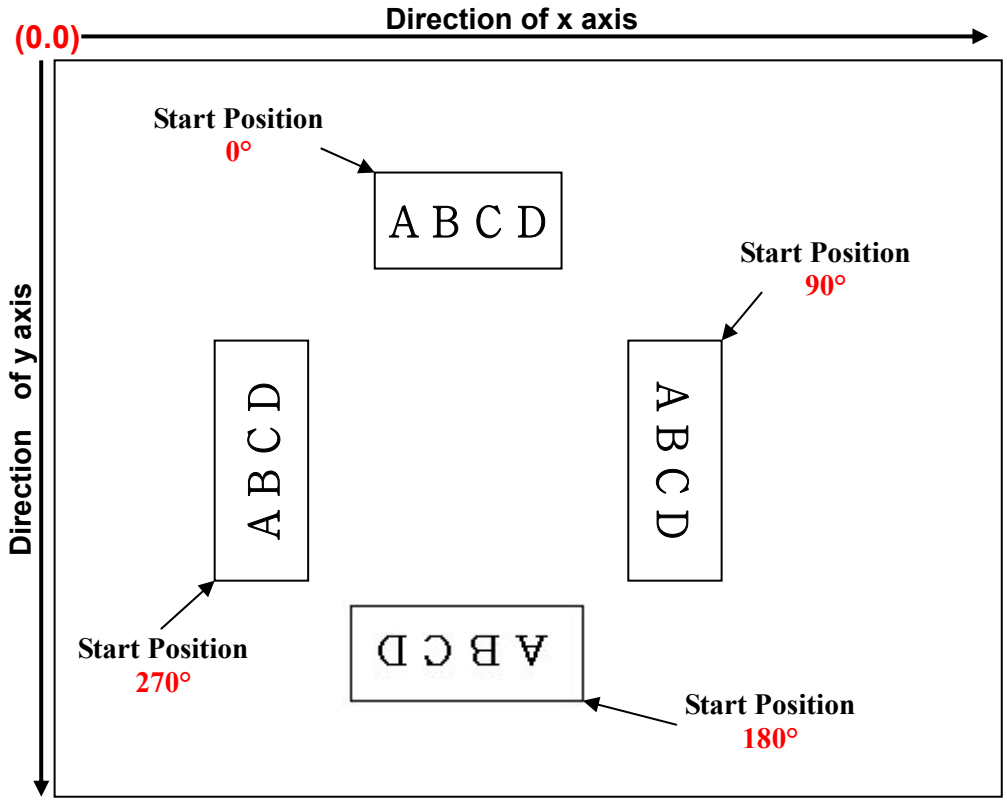
2) Font Information

Font name	Width × Height (dots)
0	09 ×15
1	12 × 20
2	16 × 25
3	19 × 30
4	24 × 38
5	32 × 50
6	48 × 76
7	22 × 34
8	28 × 44
9	37 × 58
Korean a	16 × 16(ascii:9×15)
Korean b	24 × 24(ascii:12×24)
Korean c	20 × 20(ascii:12×20)
Korean d	26 × 26(ascii:16×30)
Korean e	20 × 26(ascii:16×30)
Korean f	38 × 38(ascii:22×34)
GB2312 m	24 × 24(ascii:12×24)
BIG5 n	24 × 24(ascii:12×24)
Vector	Scalable

3) Example of text and barcode



4) Example of rotation



1-3 Command List

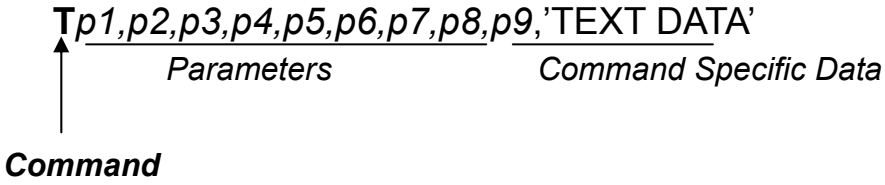
Command	Description	Remarks	Page
1. Commands for Designing Label			
T	Text	Draw text string on the image buffer	
V	Text (Vector Font)	Draw text string on the image buffer	
B1	1d barcode	Draw 1D Barcode on the image buffer	
B2	2d barcode	Draw 2D Barcode on the image buffer	
B3	Special barcode	Draw special barcode on the image buffer	
BD	Block Draw	Draw line or box on the image buffer	
CD	Circle Draw	Draw circle on the image buffer	
CS	Character Set selection	Select international code table	
P	Print	Start printing the content of image buffer	
2. Media & Buffer related Commands			
ST	Set Print Type	Select Thermal Direct / Transfer printing	
SM	Set Margin	Set the marginal value of the image buffer	
SF	Set Back-feed Option	Set back-feeding option	
SL	Set Label Length	Set length of label	
SW	Set Label Width	Set length of label	
SB	Set Buffer mode	Enable or Disable double buffering function	
CB	Clear Buffer	Clear image buffer	
3. Printer Setting Commands			
SS	Set Speed	Set printing speed	
SD	Set Density	Set printing density from level 0 to 20	
SO	Set Orientation	Set printing direction	
SP	Set serial Port	Set serial port configurations	
SA	Set Offset	Set offset value	
TA	Set Tear-off/Cut	Set Tear-off/Cut value	
4. Variable related Commands			
SC	Set Counter	Used in Template sequence	
AC	Set Counter	Used in normal mode	
SV	Set Variable	Used in Template sequence	
?	Get variables	Get content of variables and counters	
PV	Print with Variable	Use this command in Template	

Command	Description	Remarks	Page
5. Template Related Commands			
TS	Template store Start	All contents between these commands are saved in printer memory	
TE	Template store End		
TR	Template Recall	Load and reuse the stored Template	
TD	Template Delete	Delete stored Template from printer memory	
TI	Template Information	Print the list of currently stored Templates	
6. Image Data Related Commands			
IS	Image Store	PCX format image file can be stored	
IR	Image Recall	Load and reuse the stored image	
ID	Image Delete	Delete stored image	
II	Image Information	Print the list of currently stored images	
LD	Bitmap data draw	Draw bitmap image data on the image buffer	
LC	Compression bitmap data draw	Draw compression bitmap image data on specific position of image buffer	
BMP	BMP format file draw	Draw BMP format file on the image buffer	
7. Downloadable Font Related Commands			
DS	Download Bitmap font	User made Bitmap font	
DT	Download True Type font	Windows system font used	
DD	Downloadable font Delete	Delete downloaded font	
DI	Downloadable font Information	Print the list of currently stored images	
8. The Others			
@	Reset printer	Initialize the printer	
PI	Printer Information	Print current setting of printer	
CUT	Enable/Disable Cutter option	Cutting is executed after Printing is finished if cutting option is enabled by this command	
^cp	Check Printer Status	Return 2 bytes status values to host	
^cu	Check Printer Status	Return 1 byte status value to host	
^PI	Send Printer information	Send various information to host	
^MBZ	Power save mode setting	To set the power save mode	
^MBP	Transmit power save mode setting information	To send the power save mode setting values to the host	

1-4 Programming Considerations

1) All commands are case-sensitive and some commands require one or more parameters and ‘Data’.

2) Command Conventions



3) Each command line must be terminated with a ‘CR’(0Dh, 13) + ‘LF’(0Ah,10).

4) The commands which draw text, barcode, lines… just draw on the image buffer, they do not start printing. The printer will start printing when the P command comes.

! Caution

The ‘P’ command must be terminated by ‘CR’(0x0d). If not, the printer will not start printing until ‘CR’ comes.

2. Detail Description

2-1 Commands for Designing a Label

These commands are used to design a label by providing text, barcode, line, box… and to print content of image buffer on media.

1) T

Draw **Text String** on the image buffer.

2) V

Draw **Text (Vector Font) String** on the image buffer.

3) B1

Draw **1D Barcode** on the image buffer.

4) B2

Draw **2D Barcode** on the image buffer.

5) B3

Draw **Special Barcode** on the image buffer.

6) BD

Draw **Line, Block, Box & Slope** on the image buffer.

7) CD

Draw **Circle** on the image buffer.

8) CS

Set Code page and ICS(International Character Set).

9) P

Start printing the content of the image buffer.

2-1-1 T (Text String)

Description

Draw text string on the image buffer

Syntax

T*p1,p2,p3,p4,p5,p6,p7,p8,p9(,p10),***'DATA'**

Parameters

p1 : Horizontal position (X) [dot]

p2 : Vertical position (Y) [dot]

p3 : Font selection

Value	Font Size(pt)	Width × Height(dots)
0	6	9 × 15
1	8	12 × 20
2	10	16 × 25
3	12	19 × 30
4	15	24 × 38
5	20	32 × 50
6	30	48 × 76
7	14	22 × 34
8	18	28 × 44
9	24	37 × 58
a	KOREAN 1	16 × 16 (ascii 9×15)
b	KOREAN 2	24 × 24 (ascii 12×24)
c	KOREAN 3	20 × 20 (ascii 12×20)
d	KOREAN 4	26 × 26 (ascii 16×30)
e	KOREAN 5	20 × 26 (ascii 16×30)
f	KOREAN 6	38 × 38 (ascii 22×34)
m	GB2312	24 × 24 (ascii 12×24)
n	BIG5	24 × 24 (ascii 12×24)
j	Shift JIS	24 × 24 (ascii 12×24)

♣ A to Z are assigned to Downloadable font. Refer to *DS* command.

p4 : Horizontal multiplier : 1 ~ 4

p5 : Vertical multiplier : 1 ~ 4

p6 : Right-side character spacing [dot]

Plus(+)/Minus(-) option can be used. Ex) 5, +3, -10…

p7 : Rotation

Value	Rotation
0	No Rotation
1	90 degrees
2	180 degrees
3	270 degrees

p8 : Reverse printing

N : Normal printing R : Reverse printing

p9 : Bold

N : Normal B : Bold

p10 : Text Alignment(Optional)

F : p1 means the position of the first character in text string - **Left alignment**

L : p1 means the position of the last character in text string - **Right alignment**

R: Write text sting form right to left.

(BIXOLON → NOLOXIB)

♣ **This parameter is for alignment of text lines.**

'DATA' : The various data types can be used in the data field as followings.

1) Fixed text string : ' Text String'

2) Variables declared in template by **SV** command : **Vnn**

3) Counters declared by the **SC** command : **Cn**

♣ **1) , 2) and 3) can be mixed together**

Example

T50,100,3,1,1,0,0,N,N,' BIXOLON Label Printer'

T50,100,3,1,1,0,0,N,N,'Manufacturer :'*V00*

T50,100,3,1,1,0,0,N,N,*V00*

T50,100,3,1,1,0,0,N,N,'Manufacturer :'*C0*

T50,100,3,1,1,0,0,N,N,*C0*

♣ **If you want to print ' or \ then you must type like \' or \\ .**

Example

SM20,20
T26,20,0,0,0,0,0,N,N,'Font- 6 pt'
T26,49,1,0,0,0,0,N,N,'Font - 8 pt'
T26,81,2,0,0,0,0,N,N,'Font - 10 pt'
T26,117,3,0,0,0,0,N,N,'Font - 12 pt'
T26,156,4,0,0,0,0,N,N,'Font - 15 pt'
T26,200,5,0,0,0,0,N,N,'Font - 20 pt'
T26,252,6,0,0,0,0,N,N,'Font - 30 pt'
P1

Result

Font – 6 pt

Font – 8 pt

Font – 10 pt

Font – 12 pt

Font – 15 pt

Font – 20 pt

Font – 30 pt

2-1-2 V (Text String Vector Font)

Description

Draw text (Vector Font) string on the image buffer

Syntax

V*p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12,* **'DATA'**

Parameters

p1 : Horizontal position (X) [dot]

p2 : Vertical position (Y) [dot]

p3 : Font selection

U: ASCII (1Byte code)

K: KS5601 (2Byte code)

B: BIG5 (2Byte code)

G: GB2312 (2Byte code)

J: Shift-JIS (2Byte code)

a: OCR-A (1Byte code)

b: OCR-B (1Byte code)

p4 : Font width (W)[dot]

p5 : Font height (H)[dot]

p6 : Right-side character spacing [dot]

Plus (+)/Minus (-) option can be used. Ex) 5, +3, -10…

P7 : Bold

N : Normal B : Bold

p8 : Reverse printing

N: Normal printing R: Reverse printing

P9 : Text style

N: Normal I: Italic

P10 : Rotation

Value	Rotation
0	No Rotation
1	90 degrees
2	180 degrees
3	270 degrees

p11: Text Alignment (Optional)

L: p1 means the position of the first character in the text string - Left alignment

R: p1 means the position of the last character in the text string - Right alignment

C: p1 means the position of the center character in the text string - Center alignment

p12: Text string write direction

0: Write text string form left to right (BIXOLON)

1: Write text string form right to left (NOLOXIB)

'DATA': The various data types can be used in the data field as follows.

1) Fixed text string: 'Text String'

2) Variables declared in template by **SV** command: **Vnn**

3) Counters declared by the **SC** command: **Cn**

♣ **1), 2), and 3) can be mixed together.**

Example

```
V50,100,U,25,25,+1,N,N,N,0,L,0,'BIXOLON Label Printer'
V50,200,U,35,35,-1,N,N,N,0,L,0, 'Manufacturer :'V00
V50,300,U,35,35,+1,B,R,I,0,L,0, V00
V50,400,U,45,25,+1,N,N,N,0,L,0,'Vector Font Test' C0
V50,500,U,25,45,+1,N,N,N,0,L,0, C0
V50,600,a,25,45,+1,N,N,N,0,L,0, 'OCR-A font test'
```

♣ **To print ' or \, \' or \\ must be typed.**

Example

```
V50,100,U,25,25,+1,N,N,N,0,L,0,'Vector Font Test'
V50,200,U,35,35,-1,N,N,N,0,L,0,'Vector Font Test'
V50,300,U,35,35,+1,B,R,I,0,L,0,'Vector Font Test '
V50,400,U,45,25,+1,N,N,N,0,L,0,'Vector Font Test'
V50,500,U,25,45,+1,N,N,N,0,L,0,'Vector Font Test'
V50,700,U,65,65,+1,N,N,N,0,L,0,'ABCDEFGHIJKLMNO'
V50,900,U,65,65,+1,N,N,N,0,L,0,'abcdefghijklmno'
P1
```

Result

2-1-3 B1 (1 Dimensional bar code)

Description

Draw 1D Barcode on the image buffer

Syntax

B1*p1,p2,p3,p4,p5,p6,p7,p8(,p9),*'**DATA**'

Parameters

p1 : Horizontal position (X) [dot]

p2 : Vertical position (Y) [dot]

p3 : Barcode selection

p3	Barcode	p3	Barcode
0	Code39	5	UPC-A
1	Code128	6	UPC-E
2	I2of5	7	EAN13
3	Codabar	8	EAN8
4	Code93	9	UCC/EAN128

p4 : Narrow bar width [dot]

p5 : Wide bar width [dot]

p6 : Bar code height [dot]

p7 : Rotation

Value	Rotation
0	No Rotation
1	90 degrees
2	180 degrees
3	270 degrees

p8 : HRI (Human Readable Interpretation)

0 : Not printed

1 : Below the bar code(Font Size : 1)

2 : Above the bar code(Font Size : 1)

3 : Below the bar code(Font Size : 2)

4 : Above the bar code(Font Size : 2)

5 : Below the bar code(Font Size : 3)

6 : Above the bar code(Font Size : 3)

7 : Below the bar code(Font Size : 4)

8 : Above the bar code(Font Size : 4)

(p9) : quiet zone width(optional) : 0 ~ 20

♣ **Quiet zone is added to the front and end of the barcode for safe scanning. Because of the quiet zone, the barcode seems to be seen drawn in incorrect position. If p9 is not used, the printer automatically sets parameter to 0.**

Quiet zone with = p9 × narrow bar width(p4)

'**DATA'** : The various data types can be used in the data field as followings.

1) Fixed text string : ' Text String'
2) Variable declared in template by **SV** command : **Vnn**
3) Counter declared by the **SC** command : **Cn**
4) In the Code 128, when send data to printer if codeset selection commands (>A,>B,>C) will be used codeset can be selected.
   By using **>A**, Codeset will be set Codeset A.
   By using **>B**, Codeset will be set Codeset B.
   By using **>C**, Codeset will be set Codeset C.

If Codeset select command is not used, automatically set to Auto-mode.

♣ **1) , 2) and 3) can be used together**

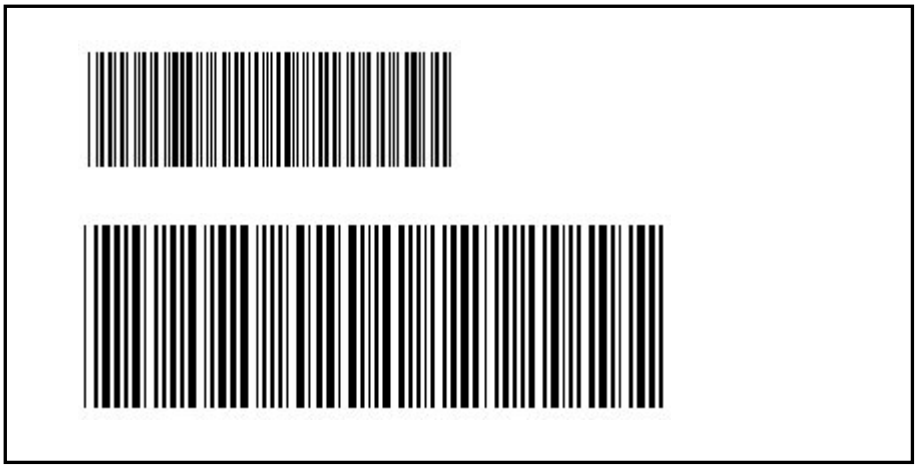
Example

B178,196,0,2,6,100,0,0,'1234567890'
B178,196,0,2,6,100,0,0,*V00*
B178,196,0,2,6,100,0,0,*C0*
B178,196,1,2,6,100,0,0,'>A1234567890'
B178,196,1,2,6,100,0,0,'>B1234567890'
B178,196,1,2,6,100,0,0,'>C1234567890>A5'

Example

SM20,20
B178,196,0,2,6,100,0,0,'1234567890' **// Caution:The position is not (178,196) but (78,196)**
B150,468,0,4,10,200,0,0,'1234567890'
P1

Result



2-1-4 B2 (2 Dimensional bar code)

Description

Draw 2D Barcode on the image buffer

Syntax

B2*p1,p2,p3……*'**DATA**'

Parameters

p1 : Horizontal position (X) [dot]

p2 : Vertical position (Y) [dot]

p3 : 2D barcode selection

p3	2D Barcode
M	MaxiCode
P	PDF417
Q	QR Code
D	Data Matrix

♣♣ Following parameters (p4, p5 …. ,Data) are barcodes-specific.
See the following pages for details of each 2D barcodes.

Maxicode(When p3 is M)

p1 : Horizontal position (X) [dot]

p2 : Vertical position (Y) [dot]

p3 : M (means 'Maxicode')

p4 : Mode selection

p4	Rotation
0	Mode0
2	Mode2
3	Mode3
4	Mode4

'DATA' : Data format is dependent on 'Mode'

Mode	Data Format
0	
2 or 3	'cl,co,pc,lpm'
4	'lpm'

cl : Class Code(3 digits)

co : Country Code(3digits)

Mode2 : Numeric Characters

Mode3 : International Characters

pc : Postal Code

lpm : Low priority message(data)

Example

1)Mode 0

B2200,200,M,*0*,'999,840,06810,7317,THIS IS A TEST OF MODE 0 STRUCTURED CARRIER MESSAGE ENCODING. THIS IS AN 84 CHAR MSG'

2)Mode 2

B2200,200,M,*2*,'999,840,06810,7317,THIS IS A TEST OF BIXOLON LABEL PRINTER SLPD220. MODE 2 ENCODING. THIS IS AN 84 CHAR.'

3)Mode3

B2200,200,M,*3*,'999,056,B1050,7317,THIS IS A TEST OF BIXOLON LABEL PRINTER SLPD220. MODE 3 ENCODING. THIS IS AN 84 CHAR.'

4)Mode4

B2200,200,M,*4*,'THIS IS A 93 CHARACTER CODE SET A MESSAGE THAT FILLS A MODE 4, UNAPPENDED, MAXICODE SYMBOL...'

PDF417(When p3 is P)

p1 : Horizontal position (X) [dot]

p2 : Vertical position (Y) [dot]

p3 : P (means 'PDF417')

p4 : Maximum Row Count : 3 ~ 90

p5 : Maximum Column Count : 1 ~ 30

p6 : Error Correction level

p6	EC Level	EC Codeword
0	0	2
1	1	4
2	2	8
3	3	16
4	4	32
5	5	64
6	6	128
7	7	256
8	8	512

p7 : Data compression method

p7	Data Type	Compression
0	Text	2 Characters per codeword
1	Numeric	2.93 Characters per codeword
2	Binary	1.2 Bytes per codeword

p8 : HRI

0 : Not Printed

1 : Below the barcode

p9 : Barcode origin point

0 : Center of barcode

1 : Upper left corner of barcode(default)

p10 : Module Width : 2 ~ 9

p11 : Bar Height : 4 ~ 99

p12 : Rotation

Value	Rotation
0	No Rotation
1	90 degrees
2	180 degrees
3	270 degrees

'DATA' : ASCII data or Binary data.

Example

B2100,750,P,30,5,0,0,1,1,3,10,0,' BIXOLON Label Printer SLPD220' **// The position is (100,750)**

QR Code(When p3 is Q)

p1 : Horizontal position (X) [dot]

p2 : Vertical position (Y) [dot]

p3 : Q (means 'QR Code')

p4 : MODEL selection

1 : MODEL1

2 : MODEL2

p5 : ECC Level

p6	Recovery Rate
L	7%
M	15%
Q	25%
H	30%

p6 : Barcode Size : 1~4

p7 : Rotation

Value	Rotation
0	No Rotation
1	90 degrees
2	180 degrees
3	270 degrees

'DATA' : ASCII data or Binary data.

Example

B2200,100,Q,2,M,4,0,'ABCDEFGHIJKLMN1234567890' **// The position is (200,100)**

Data Matrix(When p3 is D)

p1 : Horizontal position (X) [dot]

p2 : Vertical position (Y) [dot]

p3 : D (**the ECC 200 data quality format**)

p4 : Barcode Size : 1 ~ 4;

P5 : Reverse

N: Normal

R: Reverse(or Inverse) – Reverse Video or Negative image

(P6) : Rotation

Value	Rotation
0	No Rotation
1	90 degrees
2	180 degrees
3	270 degrees

'DATA' : ASCII data or Binary data.

Example

B2200,100,D,2,N,'BIXOLON Label Printer' **// The position is (200,100)**

2-1-5 B3 (Special Barcode)

Description

Draw Special Barcode on the image buffer

Syntax

B3*p1,p2,p3……*'**DATA**'

Parameters

p1 : Horizontal position (X) [dot]

p2 : Vertical position (Y) [dot]

p3 : Special barcode selection

p3	Special Barcode
I	IMB(Intelligent Mail Barcode)

♣♣ Following parameters (p4, p5 …. ,Data) are barcodes-specific.
See the following pages for details of each special barcodes.

IMB (p3 = I)

p1 : Horizontal position (X) [dot]

p2 : Vertical position (Y) [dot]

p3 : I (means 'IMB')

p4 : Rotation

Value	Rotation
0	No Rotation
1	90 degrees
2	180 degrees
3	270 degrees

P5 : HRI :

0 : Not Printed

1 : Below the barcode

'DATA' : ASCII data or Binary data.

Example

B3100,100,I,0,1,'01234567094987654321012345678 91' **// The position is (100,100)**

2-1-6 BD (Block Draw)

Description

Draw Line, Block, Box & Slope on the image buffer

Syntax

BD*p1,p2,p3,p4,p5(,p6)*

Parameters

p1 : Horizontal start position (X) [dot]

p2 : Vertical start position (Y) [dot]

p3 : Horizontal end position (X) [dot]

p4 : Vertical end position (Y) [dot]

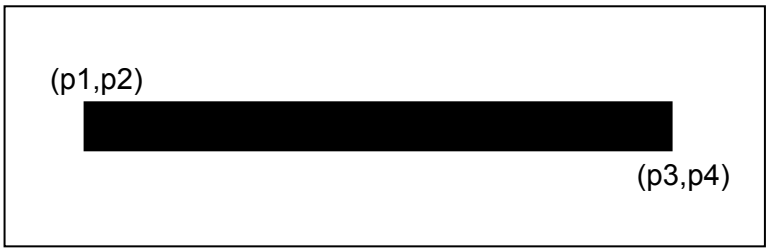
p5 : Options

p5	Type	Additional p6
O	Line Overwriting	Not necessary
E	Line Exclusive OR	Not necessary
D	Line Delete	Not necessary
S	Slope(a oblique line)	Thickness
B	Box	Thickness

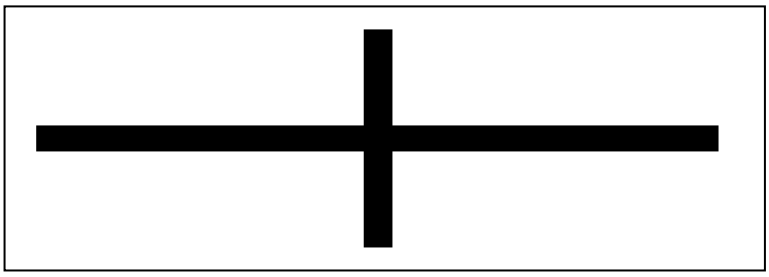
♣ If p5 is S or B, then additional p6 must follow p5.

Example

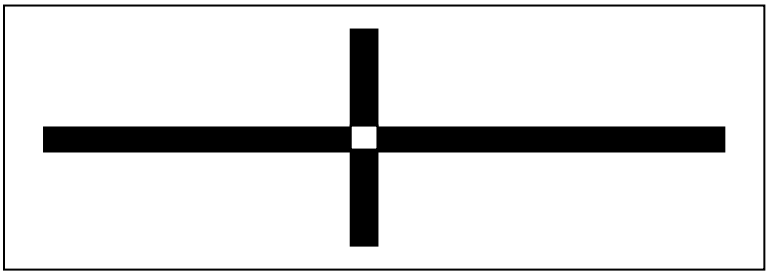
1) Start and end position



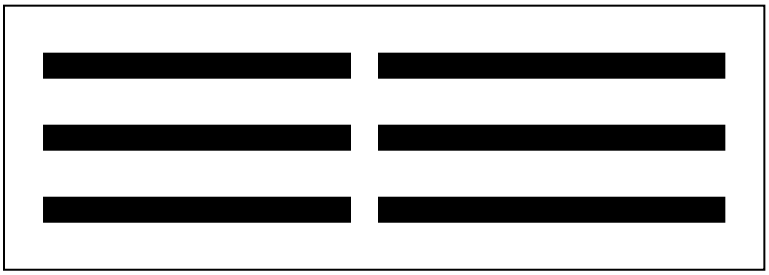
2) Overwriting mode(when p5 is O)



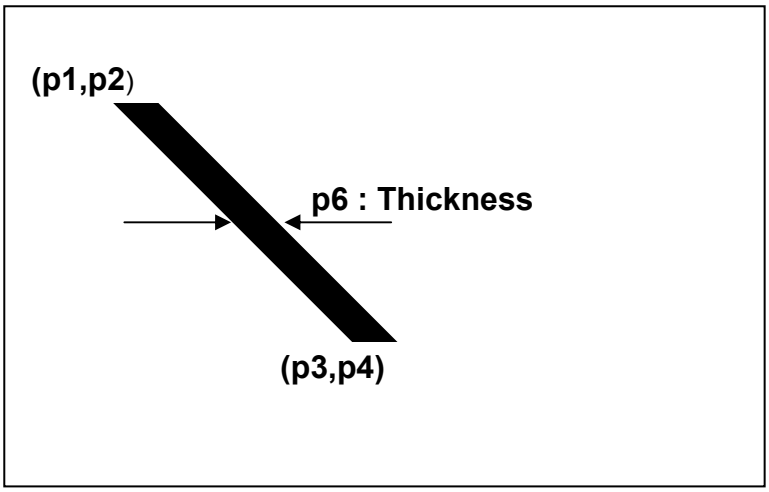
3) Exclusive OR mode(when p5 is E)



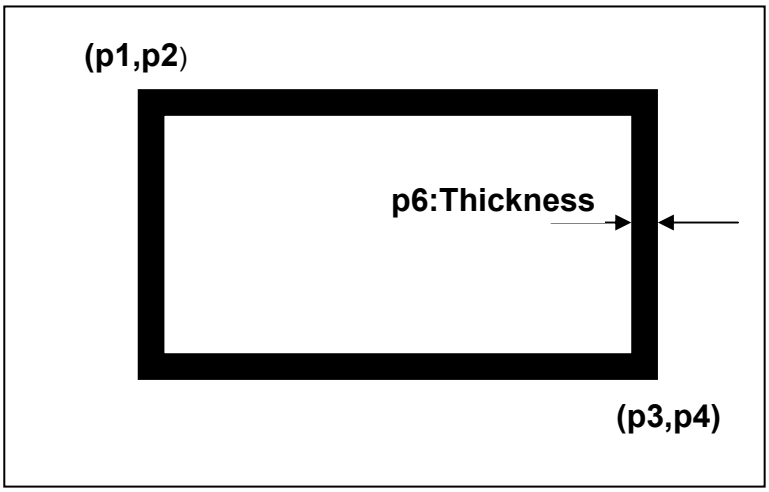
4) Delete block mode(when p5 is D)



5) Slope block mode(when p5 is S)



6) Draw box mode(when p5 is B)



2-1-7 CD (Circle Draw)

Description

Draw Circle on the image buffer

Syntax

CD*p1,p2,p3,p4*

Parameters

p1 : Horizontal start position (X) [dot]

p2 : Vertical start position (Y) [dot]

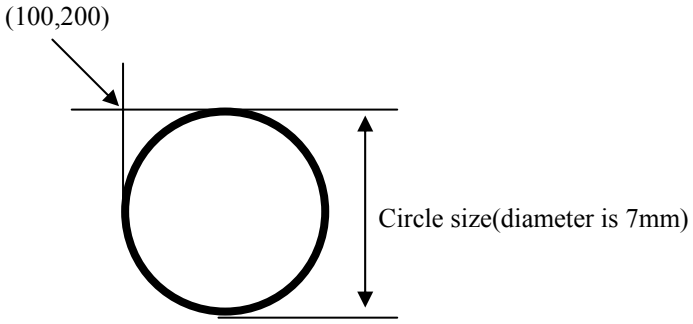
p3 : Circle Size Selection

Value	Diameter (mm)	Width × Height(dots)
1	5	40 × 40
2	7	56 × 56
3	9	72 × 72
4	11	88 × 88
5	13	104 × 104
6	21	168 × 168

p4 : Multiplier : 1 ~ 4

Example

CD100,200,2,1



2-1-8 CS (Character Set selection)

Description

To select international character set and code table

Syntax

CS*p1,p2*

Parameters

p1 : International Character Set

p1	Country
0	*U.S.A*
1	France
2	Germany
3	U.K
4	Denmark I
5	Sweden
6	Italy
7	Spain I
8	Norway
9	Denmark II
10	Japan
11	Spain II
12	Latin America
13	Korea
14	Slovenia/Croatia
15	China

p2 : Code Pages

p2	Code Table	Language
0	*CP437*	*U.S.A*
1	CP850	Latin1
2	CP 852	Latin2
3	CP 860	Portuguese
4	CP 863	Canadian French
5	CP 865	Nordic
6	WCP 1252	Latin I
7	CP 865 + WCP 1252	European Combined
8	CP 857	Turkish
9	CP 737	Greek
10	WCP 1250	Latin 2
11	WCP 1253	Greek
12	WCP 1254	Turkish
13	CP 855	Cyrillic
14	CP 862	Hebrew
15	CP 866	Cyrillic
16	WCP 1251	Cyrillic
17	WCP 1255	Hebrew
18	CP 928	Greek
19	CP 864	Arabic
20	CP 775	Baltic
21	WCP1257	Baltic
22	CP858	Latin 1 + Euro

♣ Default Setting is U.S.A standard (p1=0 and p2=0).

♣ European Combined Page

Address	Code Page
0x80	Euro Currency
0x81 ~ 0x9f	PC865
0xA0 ~ 0xff	PC1252

Country	International Character Set												
	Hex	23h	24h	40h	5Bh	5C h	5D h	5Eh	60h	7Bh	7C h	7D h	7E
	Dec	35	36	64	91	92	93	94	96	123	123	125	126
U.S.A		#	$	@	[	\	]	^	`	{	|	}	~
France		#	$	à	°	ç	§	^	`	é	ù	è	¨
Germany		#	$	§	Ä	Ö	Ü	^	`	ä	ö	ü	β
U.K.		£	$	@	[	\	]	^	`	{	|	}	~
Denmark I		#	$	@	Æ	Ø	Å	^	`	æ	ø	å	~
Sweden		#	¤	É	Ä	Ö	Å	Ü	é	ä	ö	å	ü
Italy		#	$	@	°	\	é	^	ù	à	ò	è	ì
Spain		Pts	$	@	¡	Ñ	¿	^	`	¨	ñ	}	~
Norway		#	¤	É	Æ	Ø	Å	Ü	é	æ	ø	å	ü
Denmark II		#	$	É	Æ	Ø	Å	Ü	é	æ	ø	å	ü
Japan		#	$	@	[	¥	]	^	`	{	|	}	~
Spain II		#	$	á	¡	Ñ	¿	é	`	í	ñ	ó	ú
Latin America		#	$	á	¡	Ñ	¿	é	ü	í	ñ	ó	ú
Korea		#	$	@	[	\	]	^	`	{	|	}	~
Slovenia/Croatia		#	$	Ž	Š	Đ	Ć	Č	ž	š	đ	ć	č
China		#	¥	@	[	\	]	^	`	{	|	}	~

ASCII Code		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	0 0	Control Characters															
	1 16																
	2 32		!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
	3 48	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
0~31 : Control Code	4 64	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
32~127 : Alphanumeric	5 80	P	Q	R	S	T	U	V	W	X	Y	Z	[	]	^	_	`
	6 96	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p
	7 112	q	r	s	t	u	v	w	x	y	z	{	|	}	~	⌂	

※ Refer to the "Code Pages Manual" for more extension code pages table.

2-1-9 P (Print)

Description

Let the printer start printing the content of image buffer

Syntax

P*p1,[p2]*

Parameters

p1 : Number of label sets : 1 ~ 65535

p2 : Number of copies of each label : 1 ~ 65535

♣ The P command cannot be used in a template sequence. If printing command is needed in template sequence, then use the PV command(See the example of next page).

! Caution

The 'P' command should be terminated by 'CR'(0x0d). If not, the printer will not start to print until 'CR' comes.

Example

(1) In case of Using P (P is used outside of template sequence)

```
TS'TPL_TST1'                          // Start Template Store
SV00,15,N,'Model Name :'              // Declare variable V00
T50,100,3,1,1,0,0,N,N,'Model Name :'V00  // T command with variable
TE                                    // End Template Store

TR'TPL_TST1"                          // Recall stored template 'TPL_TST1'
?                                     // Get content of variable used in recalled template
SLPD220                               // Content of variable V00
P3,2                                  // when using P command, It must not be inside template,
                                      // but be used after recalling the template and entering the
                                      // contents of all variables.
                                      // After P command, printer starts printing.
```

(2) In case of Using PV(PV is used inside of template sequence

```
TS'TPL_TST1'                          // Start Template Store
SV00,15,N,'Model Name :'              // Declare variable V00
SV01,2,N,'# of set :'                 // Declare variable V01
SV02,2,N,'# of copies : '             // Declare variable V02
T50,100,3,1,1,0,0,N,N,'Model Name :'V00  // T command with variable
PVV01,V02                             // PV command can be used inside the template
TE                                    // End Template Store

TR'TPL_TST1"                          // Recall stored template 'TPL_TST1'
?                                     // Get content of variable used in recalled template
SLPD220                               // Content of variable V00
3                                     // Content of variable V00
2                                     // Content of variable V00
                                      // As soon as all contents of variables are entered'
                                      // printer will starts printing
```

2-2 Media & Buffer Related Commands

1) ST

Select Thermal Direct/Transfer Printing.

2) SM

Set marginal value in label(Image buffer).

3) SF

Set back-feed option.

4) SL

Set label(Image buffer) length.

5) SW

Set label(Image buffer) width.

6) SB

Set buffer mode(Enable or disable Double Buffering).

7) CB

Clear Image Buffer.

2-2-1 ST (Set Printing Type)

Description

Select Thermal Direct Printing or Thermal Transfer Printing

Syntax

ST*p1*

Parameters

p1 : Direct Thermal / Thermal Transfer

- d : Direct Thermal
- t : Thermal Transfer

2-2-2 SM (Set Margin)

Description

Set marginal value of the image buffer

This command moves the origin point (0,0) to (p1,p2) and make (p1,p2) become the new origin.

Syntax

SM*p1,p2*

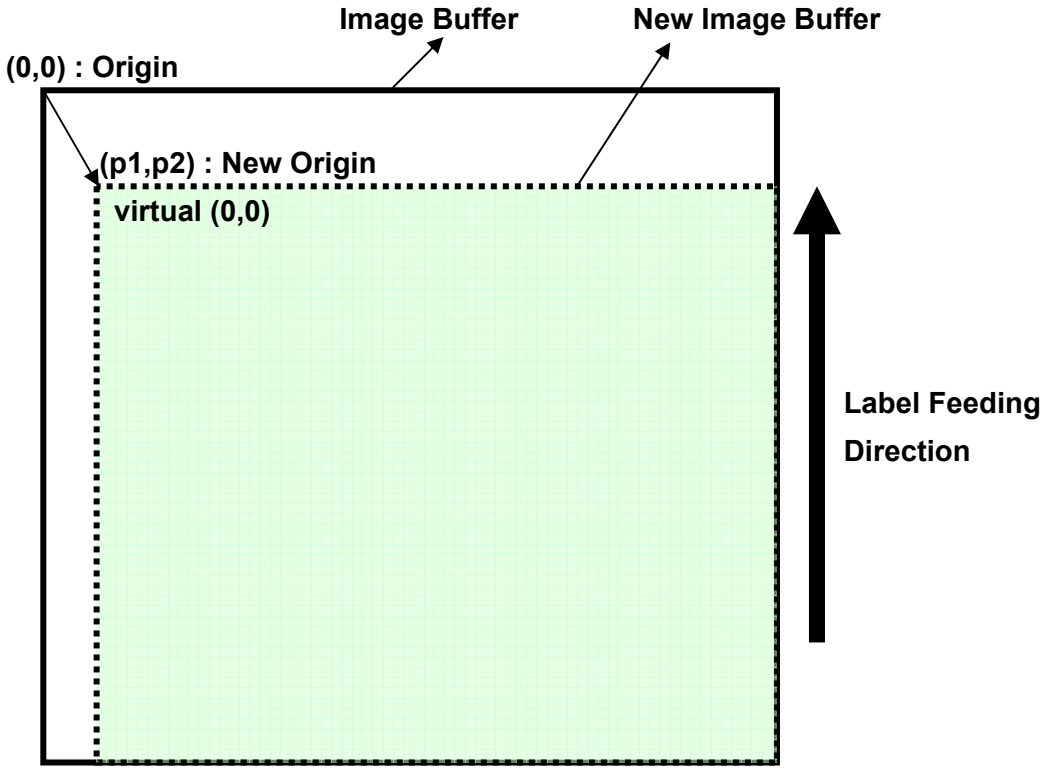
Parameters

p1 : Horizontal margin [dots]

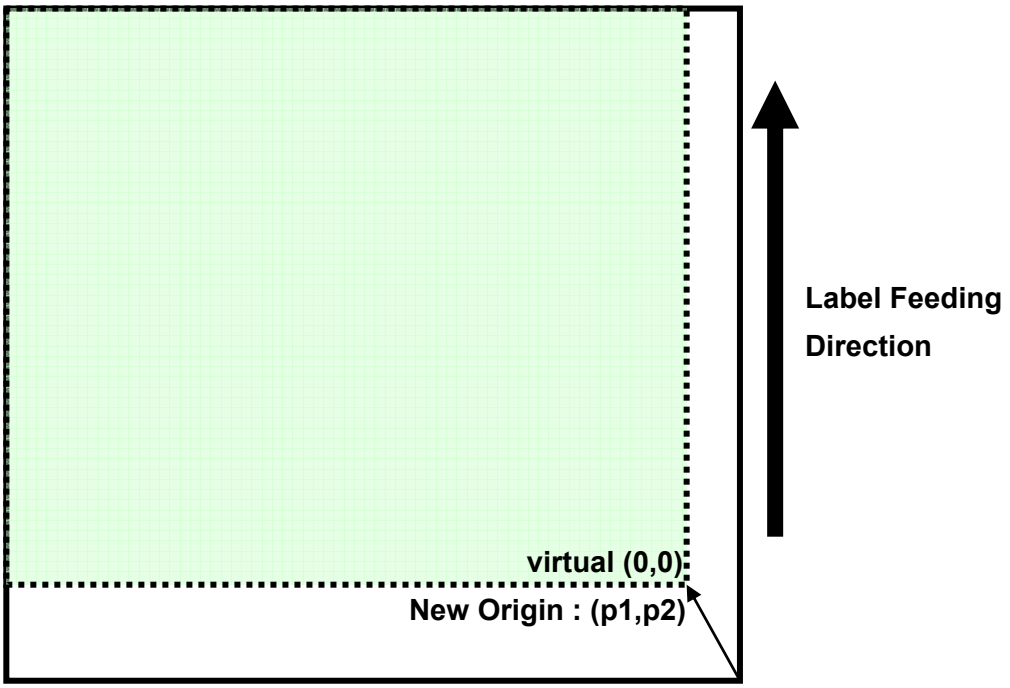
p2 : Vertical margin [dots]

♣ The origin point is upper-left point of the image buffer

** When printing orientation is from top to bottom



** When printing orientation is from bottom to top.



2-2-3 SF (Set Back-Feed Option)

Description

Set back-feed option

This command decides whether printer does back-feed action before starting printing.

Syntax

SF*p1(,p2)*

Parameters

p1 : Enable/Disable

- 0 : Disable back-feed option.
- 1 : Enable back-feed option(Default)

p2 : Back feeding step quantity.

- This parameter is valid when p1 is 1.
- The step quantity defined by user can't exceed printer's default feeding quantity.
- 0 means printer's default feeding quantity.

♣ This option is useful for the continuous paper or black mark media with perforation line away from black mark.

♣ The printer's default back feeding step quantity depends on the printer models and printer modes such as normal, peeler or cutter.

Examples)

SF0 → Disable Printer's back-feeding option.

SF1 → Default quantity of Back feed is executed before printing.

SL1,0 → Default quantity of Back feed is executed before printing.

SL1,100 → 100 step's Back feed is executed before printing.

SL0,100 → Back feed is disabled and p2(100) is ignored.

2-2-4 SL (Set Length)

Description

Set length of label and gap(or Black Mark) and specify media type

Syntax

SL*p1,p2(,p3)(,p4)*

Parameters

p1 : Label length [dots] : Maximum 2432 dots(12 inch)

- ♣ **Double buffering feature can be used only when label length(p1) is less than 1216(2432/2, 6inch) dots.**
- ♣ **If p1 is over 1216 dots, the double buffering feature will be automatically released.**
- ♣ **So if you don't use double buffering feature, you can design maximum 2432 dots(12 inch) size label.**

p2 : Gap length or thickness of black line [dots]

p3 : Media Type

p3	Media type
G	*Gap*
C	Continuous
B	Black Mark

- ♣ **If this parameter is not used, automatically set to G(Gap type).**
- ♣ **The default value of label length is 6 inch(1216 dots)**
- ♣ **This command sets the length of image buffer and the printer will print and form feed as much as the length set by this command.**
- ♣ **When using Continuous type media, the label length must be set.**

p4 : Offset Length between Black Mark(or Gap) and perforation line [dots]

- ♣ **This parameter is valid when p3 parameter is used.**

Examples)

SL1200,20 → Gap media,Media length:1200dots,Gap length:20dots

SL1200,20,C → Continuuous media,Media length:1200dots,Gap length:20dots

SL1200,20,G → Gap media,Media length:1200dots,Gap length:20dots

SL1200,20,B → Black Mark media,Media length:1200dots,Gap length:20dots

The perforation line is on the black mark.

SL1200,20,B,200→ Black Mark media,Media length:1200dots,Gap length:20dots

The perforation line is 200 dots behind from black mark.

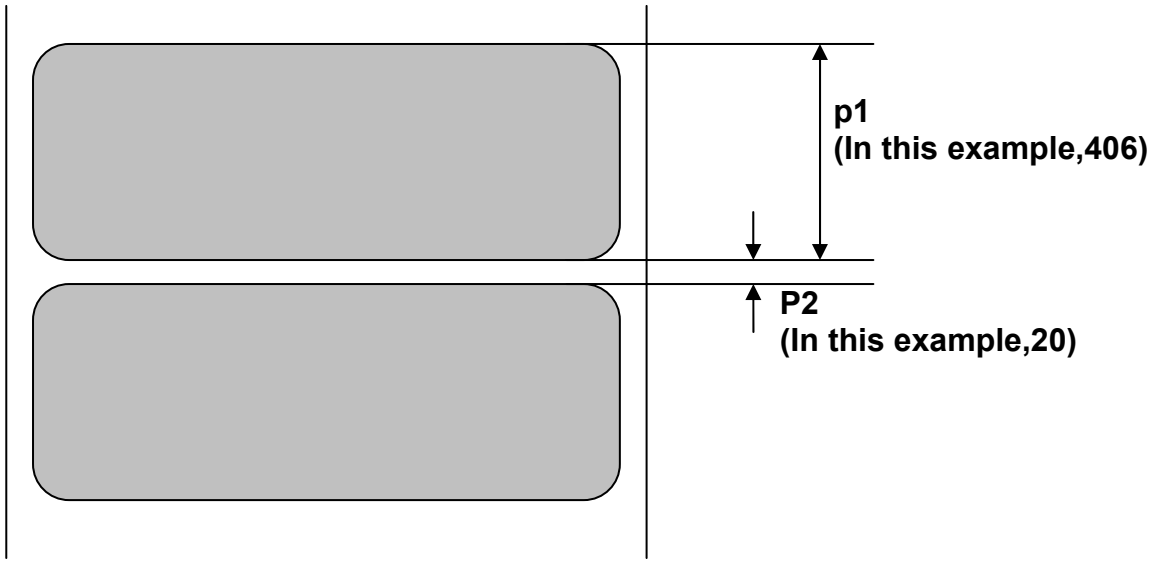
♣ **In the Gap Mode, the printer will form feed until meeting the next gap.**

♣ **In the Continuous Mode, the printer will form feed as much as label length set by SL.**

♣ **In the B/M Mode, the printer will form feed until meeting the next B/M.**

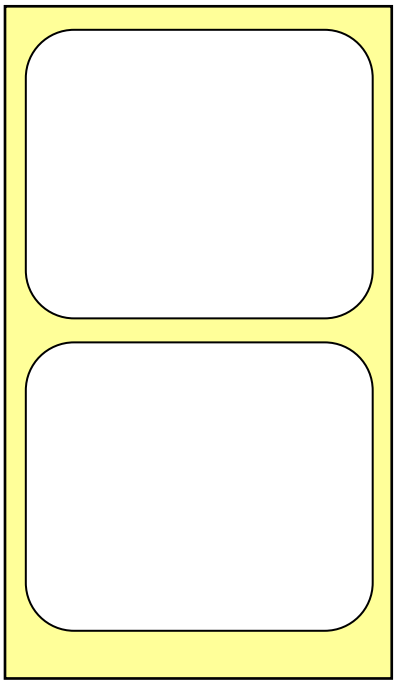
Example – p1 & p2(Length)

SL406,20 // Set label length to 406 dots (2 inch, 50mm) and gap length to 20 dots(2.5mm)

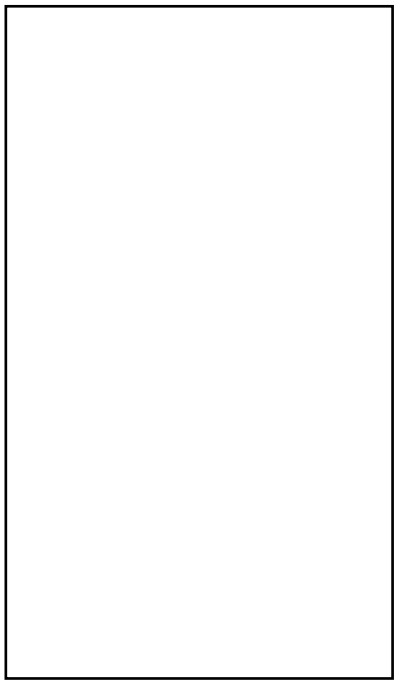


Example – p3(Media Type)

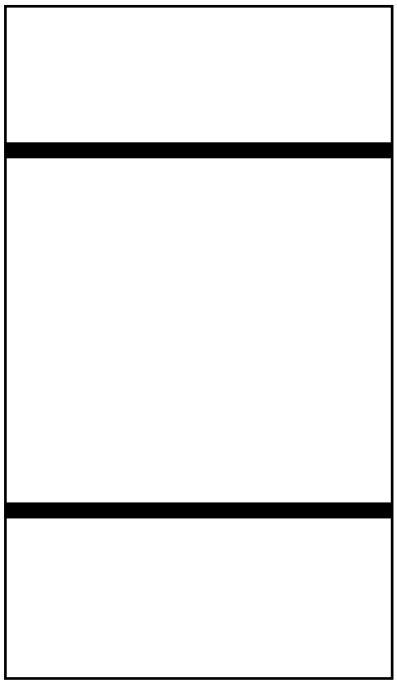
1. Gap Type



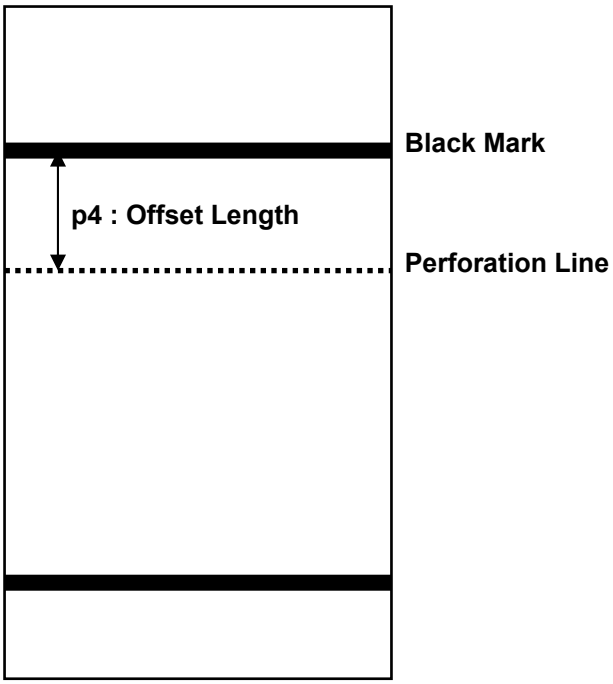
2. Continuous Type



3. Black Mark Type



Example – p4(Offset Lenght)



2-2-5 SW (Set Width)

Description

Set label width

Resize the image buffer to match the label size.

Syntax

SW*p1*

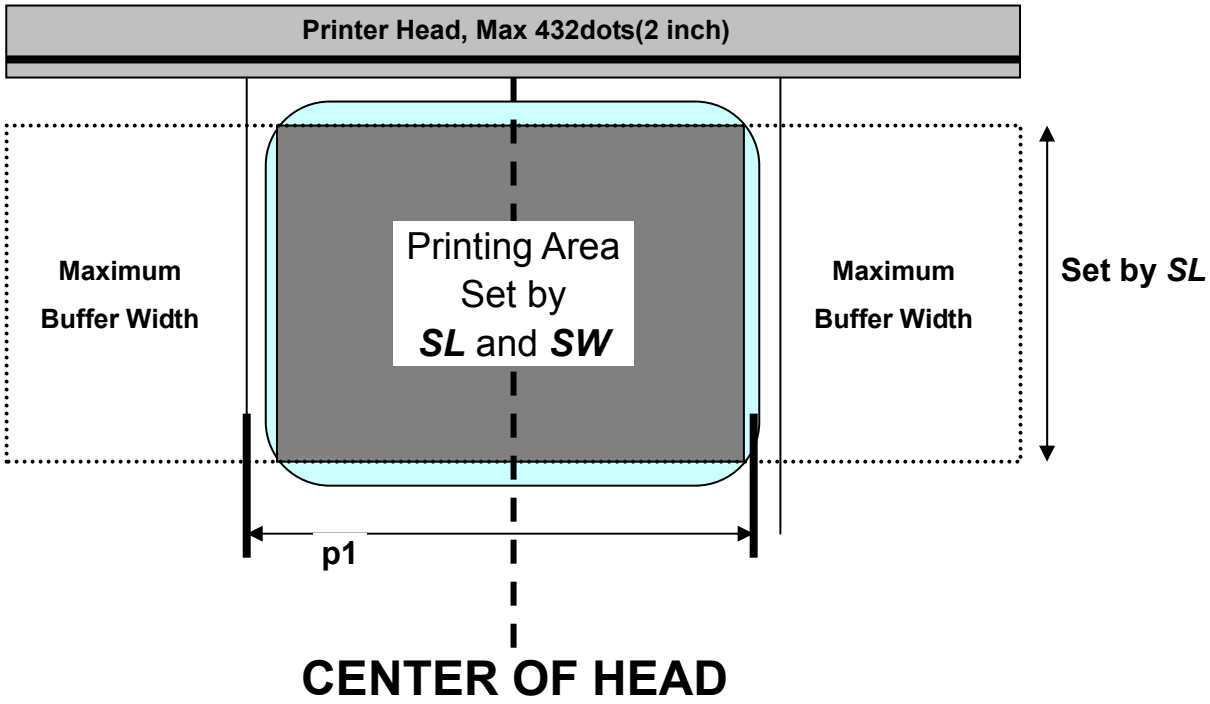
Parameters

p1 : Label width [dots]

- **The default value of label width is 2 inch(432 dots) and that is the maximum printable width.**
- **SLP-D220 is the center aligned printer and media is positioned in the center of the head.**

Example

SW432 // Set label width to 2 inch(432 dots)



2-2-6 SB (Set Buffer mode)

Description

Set double buffer mode

Syntax

SB*p1*

Parameters

p1 : Enable **'Double Buffering'** function.

0 : Disable double buffer mode

1 : Enable double buffer mode(Default)

♣ **Double buffering feature enables the printer to construct the image buffer for the next label while printing the current label.**

♣ **Double buffering feature can be used only if the label length set by SL is less than half of the maximum label length.**

2-2-7 CB (Clear Buffer)

Description

Clear image buffer and be ready to make a new label

Syntax

CB

Example

CB // Clear Image Buffer

2-3 Printer Setting Commands

1) SS

Set printer speed

2) SD

Set printing density

3) SO

Set printing orientation

4) SP

Set serial port

5) SA

Set Offset

6) TA

Set Tear-off/Cut

2-3-1 SS (Set Speed)

Description

Set print speed

Syntax

SS*p1*

Parameters

p1 : Speed set value

Value	Speed
0	2.5 ips
1	3.0 ips
2	4.0 ips
3	5.0 ips
4	6.0 ips
5	7.0 ips
6	8.0 ips

2-3-2 SD (Set Density)

Description

Set printing density

Syntax

SD*p1*

Parameters

p1: Density Level

- 0 ~ 20 (0 is the lowest density)

2-3-3 SO (Set Orientation)

Description

Set printing direction

Syntax

SO*p1*

Parameters

p1 : Printing direction

T : Print from top to bottom(default)

B : Print from bottom to top

Example

SOT // Print from top of the image buffer to bottom.

SOB // Print from bottom of the image buffer to top.

Printing Direction

1. SOT (Print from Top to Bottom) **2. SOB (Print from Bottom to Top)**

2-3-4 SP (Set Port)

Description

Set serial port

Syntax

SP*p1,p2,p3,p4*

Parameters

p1 : Baud rate

Value	Baud Rate(bps)
0	9,600
1	19,200
2	38,400
3	57,600
4	115,200

p2 : Parity

Value	Parity
O	Odd parity
E	Even parity
N	***No parity(Default)***

p3 : Number of data bits

Value	Data bits
7	7 bit
8	***8 bits (Default)***

p4 : Number of stop bits

Value	Stop bits
1	***1 bit(Default)***
2	2 bits

2-3-5 SA (Set Offset)

Description

Save (set) offset length between black marks (or gap) and dotted lines [dots]

Syntax

SA*p1*

Parameters

p1 : -100~100

♣ Offset values saved via the use of SA commands are stored permanently on the printer. (Offset values saved via the cf. SL command are reset after the power is turned off.)

2-3-6 TA (Tear-off/Cutter Position Setting)

Description

This function regulates the label cutting location After printing

Tear-off position or Cut position can adjust

Syntax

TA*p1*

Parameters

p1 : -100~100

♣ Tear-off/Cutter Position values saved via the use of TA commands are stored permanently on the printer.

2-4 Variable Related Commands

1) SC

Counters which is used in template sequence

2) AC(Auto Counter)

Counters which is used in normal commands sequence
(outside of template sequence)

3) SV

Set variable

4) ?

Get data for counter and variable

5) PV

Print with variables

2-4-1 SC (Set Counter)

Description

Define one counter of total 10 counters

Counters must be used in Template sequence and execute consecutive auto-numbering function.

Syntax

SC*p1,p2,p3,p4*,**'Prompt'**

Parameters

p1: Identity of Counter : 0 ~ 9

♣ **Total 10 counters, from C0 to C9, are provided.**

p2: The size of the field which displays the content of counter : 1 ~ 27

p3: Justification in field(Field size is p2)

Value	Justification
N	No
R	Right
L	Left
C	Center

p4 : Step Value : ±1 ~ ±9

♣ **+ or – symbol must precede . Ex) –2 or +3**

'Prompt': This text string is transmitted to host(PC) by serial interface in order to give information to host about the declared counter.

♣ **The data field of T(Text) or B(Barcode) commands is used to print the contents of counter.**

♣ **SC should be used just in Template sequence. If you want to use counter function in normal mode(not in Template), use the AC(Auto Counter).**

Example

SC0,7,N,+3,'Please Enter Serial Number'

2-4-2 AC (Auto Counter)

Description

Define one counter of total 10 counters

Counters can be used in normal mode(not in Template) and execute consecutive auto-numbering

Syntax

AC*p1,p2,p3*,**'Start Value'**

Parameters

p1 : Identity of Counter : 0 ~ 9

♣ **Total 10 counters, from C0 to C9, are provided.**

p2 : The size of the field which displays the content of counter : 1 ~ 27

p3 : Step Value : ±1 ~ ±9

♣ **+ or – symbol must precede . Ex) –2 or +3**

'Start Value' : Start value of auto-counting. Just digits can be used in this field

---

♣ **The Auto-counter defined by AC command can be printed with T and B1 command.**

♣ **This function is useful to print serial number or serial barcode without using Template.**

♣ **AC can not be used in Template sequence. If you want to use counter function in Template sequence, use the SC command.**

Example

AC0,3,+1,'123' **// Please input the start value of counting between ' marks**

AC1,7,+1,'1234567'

T100,100,3,1,1,0,0,N,N,C0

B1100,400,0,2,7,100,0,1,12,C1

P3,1

2-4-3 SV (Set Variable)

Description

Define variables for the text or barcode ‘data’ fields

Syntax

SV*p1,p2,p3,***’Prompt’**

Parameters

p1 : Identity of Variables : 00 ~ 99

p2 : Maximum number of characters : 1 ~ 99

p3 : Justification in field(Field size is p2)

Value	Justification
N	No
R	Right
L	Left
C	Center

‘Prompt’ : This ASCII text field is used to ask a value to be entered for the variable(p1) and is transmitted to the host by serial interface.

♣ The data field of T(Text) or B(Barcode) commands is used to print the contents of variable.

♣ Variable is entered to data field like V00 or V01.

Example

SV01,20,N,’Please Enter Product Code :’

2-4-4 ? (Get Variables)

Description

Use this command to get the content of variables or counters

Syntax

?

Content of variable

♣ Data must be entered in ascending order

Example

TS'Template1'	// Template Store Start
SV00,20,N'Enter Company Name : '	// Declare(Set) variable V00
SV01,15,N'Enter Product Code :'	// Declare(Set) variable V01
T50,30,3,1,1,0,0,N,N,V00	// Use T command to print V00
T50,150,3,1,1,0,0,N,N,'Code : 'V01	// Use T command to print V01
TE	// Template Store End

TR'Template1'	// Recall Template1
?	// Start to get data for variables
SEM	// data for V00
D220	// data for V01
P1	**// Start Printing when the P command comes**

Result

SEM
Code : D220

2-4-5 PV (Print with Variables)

Description

This command is used in template sequence.

The parameters are given by variables.

Syntax

PV*p1,[p2]*

Parameters

p1 : Number of label sets : 1 ~ 65535

p2 : Number of copies of each label : 1 ~ 65535

Example

TS'Template1'	// Template Store Start
SV00,20,N,'Please Input the Name :'	// Declare(Set) variable V00
SV01,5,N,'Input Number of label sets :'	// Declare(Set) variable V01
SV02,5,N,'Input Number of label copies :'	// Declare(Set) variable V02
T50,30,3,1,1,0,0,N,N,V00	// Write V00 to image buffer
PV*V01,V02*	// Print V00, V02 copies, V01 sets
TE	// Template Store End
TR'Template1'	// Recall Template1
?	// Start to get data for variables
This is PV Test	// data for V00
2	// data for V01
1	// data for V02

***** Start Printing as soon as data for all variables(and counters) are entered. *****

2-5 Template Related Commands

Template(a certain format of label, sequence of SLCS commands) related commands

1) TS

Indicate start of template sequence store.

2) TE

Indicate end of template sequence store.

3) TR

Recall and reuse stored template.

4) TD

Delete stored template.

5) TI

Print the list of all templates stored in memory.

2-5-1 TS (Template store Start)

Description

Start template sequence storing.

All the contents following ‘TS’ are stored in memory until meeting ‘TE’ Command.

Syntax

TS’*Template name’*

Parameters

‘Template name’: This name will be used when ‘Recall’ the stored template.

- ♣ The name is allowed to be up to 10 characters long.
- ♣ The ‘Template name’ is **Case-Sensitive.**

♣ TI command shows the list of currently stored Templates.

2-5-2 TE (Template store End)

Description

End template sequence storing

Syntax

TE

♣ When storing is finished, the printer sends ‘!’ to the host to prompt end of storing.

Example

```
TS’Template1’                    // Start template storing
……
TE                               // End template storing
```

2-5-3 TR (Template Recall)

Description

Recall the stored template from memory to make a label and print that

Syntax

TR*'Template name'*

Parameters

'Template name' : Indicate the template to be recalled.

- ♣ The name is allowed to be up to 10 characters long.
- ♣ The 'Template name' is **Case-Sensitive.**

Example

TR'Template1' // Recall 'Template1'

- ♣ **If recalled Template does not include any variable or counter, just 'P' command is enough to start printing.**
- ♣ **If recalled Template includes variables or counters but not 'PV'(Print with Variables), use '?' command to get data for variables and counters and finally 'P' command is necessary to start printing.**
- ♣ **If recalled Template includes PV commands, printing will start as soon as all data for variables and counters are entered.**

2-5-4 TD (Template Delete)

Description

Delete stored template from memory

Syntax

TD*'Template name'*

Parameters

'Template name' : Indicate the template to be deleted.

- ♣ The name is allowed to be up to 10 characters long.
- ♣ The 'Template name' is **Case Sensitive**.
- ♣ By using *****, all templates will be deleted from memory.

Example

TD'Template1' // Delete 'Template1'

TD* // Delete all currently stored templates

2-5-5 TI (Template Information)

Description

Print list of currently stored templates and available memory space

Syntax

TI

Example

TI

Result

```
Templates Information
=======================
1. Template1
2. Template2
Available template memory : 53Kbyte
```

2-6 Image Related Commands

These commands provide functions to download and print graphic data.
PCX and BMP format file are supported and bitmap image data can be printed directly.

1) IS

Download PCX format image data to NV(Non Volatile) area of memory.

2) IR

Recall and print downloaded image data.

3) ID

Delete image data in NV memory.

4) II

Print all images stored in memory.

5) LD

Draw the bitmap image data directly on specific position on image buffer.

6) LC

Draw compression bitmap image data on specific position of image buffer.

7) BMP

Draw BMP format image file directly on specific position on image buffer.

2-6-1 IS (Image Store)

Description

Download PCX format Image file into the Printer Memory

Syntax

IS*p1*,**'Image name'DATA OF *.PCX**

Parameters

p1 : The size of image file in unit of byte.

'Image name' : This is the name that will be used when recalling the stored image data.

- ♣ The name is allowed to be up to 10 characters long.
- ♣ The name is case sensitive.

DATA OF *.PCX : Binary data string of PCX file.

2-6-2 IR (Image Recall)

Description

Recall the stored image from memory and draw that on the image buffer

Syntax

IR*p1*,*p2*,**'Image name'**

Parameters

p1 : Horizontal position (X) [dot]

p2 : Vertical position (Y) [dot]

'Image name' : Indicate the image data to be recalled.

- ♣ **Variable can be used in this field.**
- ♣ **The name is allowed to be up to 10 characters long.**
- ♣ **This name is Case Sensitive.**

Example

IR30,100,'Image1' // Recall 'Image1'

IR30,100,**V01** // Variable can be used in name field

2-6-3 ID (Image Delete)

Description

Delete stored image from memory

Syntax

ID*'Image name'*

Parameters

'Image name' : Indicate the Image in memory to be deleted.

- ♣ The name is allowed to be up to 10 characters long.
- ♣ This name is **Case Sensitive**.
- ♣ By using *, all images in memory will be deleted.

Example

ID'Image1' // Delete 'Image1'

ID* // Delete all currently stored images

2-6-4 II (Image Information)

Description

Print list of currently stored images in memory and available memory space

Syntax

II

Example

II

Result

```
Image Information
=====================
1. Image1
2. Image2
Available Images memory : 5.3Kbyte
```

2-6-5 LD

Draw bitmap image data on specific position of image buffer

Syntax

LDxL xH yL yH dhL dhH dvL dvH d1~dk

Parameters

xL : Low byte of horizontal start position (X) [dot]
xH : High byte of horizontal start position (X) [dot]
→ Start position in x direction = xH * 256 + xL

yL : Low byte of vertical start position (Y) [dot]
yL : High byte of vertical start position (Y) [dot]
→ Start position in y direction = yH * 256 + yL

dhL : Low byte of the number of bytes in x-direction.
dhH : High byte of the number of bytes in x-direction.
→ Number of data in x direction = dhH * 256 + dhL

dvL : Low byte of the number of lines.
dvH : High byte of the number of lines.
→ Number of data in y direction = dvH * 256 + dvL

d1~dk : bitmap image data.
→ k = (dhH*256 + dhL) * (dvH*256 + dvL)

! CAUTION

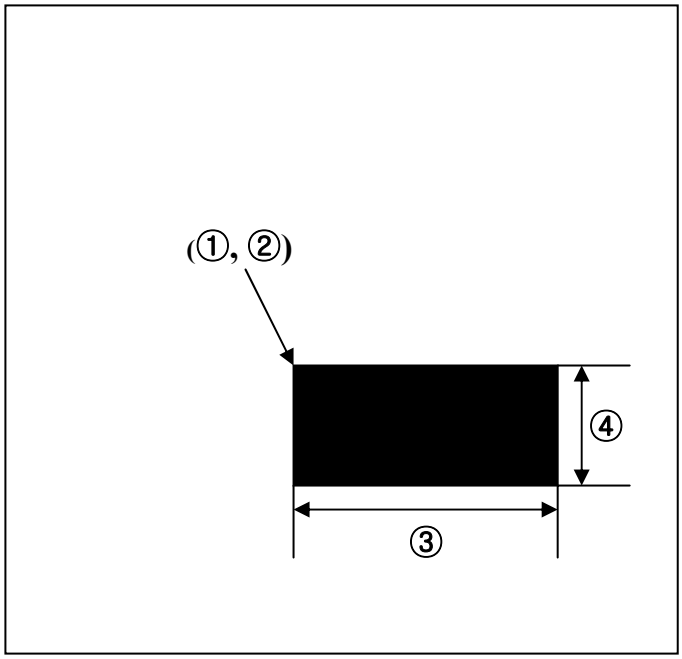
There are no commas(,) and no space between each parameters.

Example

LD 0x11 0x02 0x40 0x02 0x08 0x00 0x20 0x00 0xFF ~ 0xFF

① ② ③ ④ ⑤

① x position : 0x02 * 0x100(256) + 0x11 = 0x211(529)
② y position : 0x02 * 0x100(256) + 0x40 = 0x240(576)
③ horizontal data number : 0x00 * 0x100(256) + 0x08 = 0x08(8)
④ vertical data number : 0x00 * 0x100(256) + 0x20 = 0x20(32)
⑤ bitmap data : total number = 8 * 32 = 256



2-6-6 LC

Draw compression bitmap image data on specific position of image buffer

Syntax

LCp1p2xL xH yL yH dhL dhH dvL dvH d1~dk

Parameters

p1: Compression type
R: RLE
p2: Color
0x00: black
0x01: Color(red or blue)
xL : Low byte of horizontal start position (X) [dot]
xH : High byte of horizontal start position (X) [dot]
→ Start position in x direction = xH * 256 + xL

yL : Low byte of vertical start position (Y) [dot]
yH : High byte of vertical start position (Y) [dot]
→ Start position in y direction = yH * 256 + yL

dhL : Low byte of the number of bytes in x-direction.
dhH : High byte of the number of bytes in x-direction.
→ Number of data in x direction = dhH * 256 + dhL

dvL : Low byte of the number of lines.
dvH : High byte of the number of lines.
→ Number of data in y direction = dvH * 256 + dvL

d1~dk : Compression bitmap image data.
→ k = (dhH*256 + dhL) * (dvH*256 + dvL)

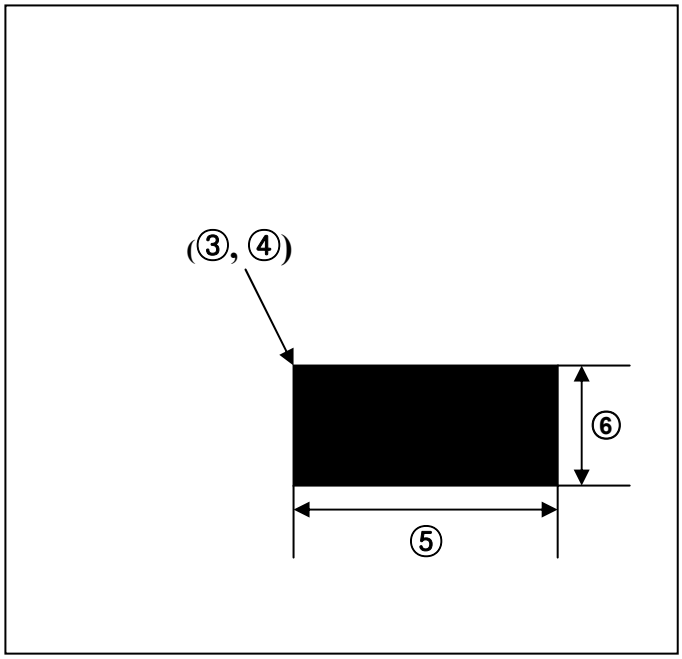
! CAUTION
There are no commas(,) and no space between each parameters.

Example

LC R 0x00 0x11 0x02 0x40 0x02 0x08 0x00 0x20 0x00 0xFF ~ 0xFF

① ② ③ ④ ⑤ ⑥ ⑦

①Compression type : R =RLE
②Color : 0x00 = Black
③x position : 0x02 * 0x100(256) + 0x11 = 0x211(529)
④y position : 0x02 * 0x100(256) + 0x40 = 0x240(576)
⑤horizontal data number : 0x00 * 0x100(256) + 0x08 = 0x08(8)
⑥vertical data number : 0x00 * 0x100(256) + 0x20 = 0x20(32)
⑦bitmap data : total number = 8 * 32 = 256



RLE compression

This is the algorithm to compress the continuous data.

The compression is applied to 0x00 & 0xff data but not the others.

0xff 0x04 data is created if 0xff is repeated four times like 0x00 0x00 0x00 0x00.

In the same way, 0x00 0x04 is created by four times repeats of 0x00 such as 0x00 0x00 0x00 0x00.

Here is the example of compression.

예) 0x78 0x78 0xff 0xff 0xff 0xff 0xff 0x22 0x00 0x00 0x00 0x00

0x78 0x78 0xff 0x05 0x22 0x00 0x04

2-6-7 BMP

Send BMP format file directly to printer.
Just white/black BMP file is supported

Syntax

BMPp1,p2↓
Data string of *.bmp

Parameters

p1 : Horizontal position (X) [dot]
p2 : Vertical position (Y) [dot]

---

1. ↓ means 'CR(+LF)'
2. There is comma(,) between p1 and p2.
3. After p2(Before sending BMP data string) 'CR(+LF)' must follow.

Example

In dos mode,
COPY bmp.txt+image2.bmp+P.txt LPT1 /b

Bmp.txt

BMP200,200 ↓

P.txt

P1 ↓

2-7 Downloadable font Related Commands

Download fonts into the printer memory. Users can download special size or special design of ASCII font and use this font with T command.

1) DT

Download True Type Font into Printer Memory

2) DD

Delete downloaded fonts from memory

3) DI

Print all downloaded fonts in memory and available memory space

2-7-1 DT (Download True Type Font)

Description

Download windows system font into printer memory

Syntax

DT*p1,p2,***'Font Name' a_1b_1(DATA$_1$)a_2b_2(DATA$_2$)…a_nb_n(DATA$_n$)**

Parameters

p1 : Total number of characters to be downloaded: 0~255

p2 : Font Height: 0~255

Font name : A ~ Z

a_n : Character position in ASCII Table(0~255)

b_n : Font width(dots)

(DATA$_n$) : Character Bitmap Data

Total bytes of bitmap data : **p2** × (**b_n**+7)/8 bytes

Example

44 44 2a 0d 0a 44 54 60 14 27 41 27 DD*..DT..'A'

p1 p2 Font name

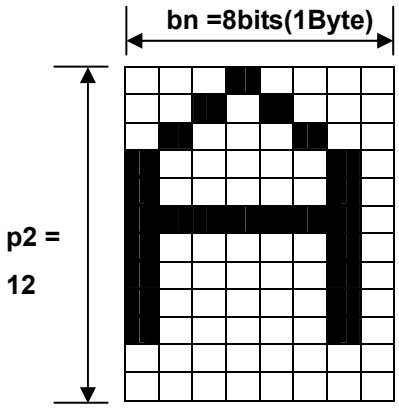
20 0b ……

a_1 b_1 DATA$_1$: p2×(**b_1**+7)/8 bytes

21 0c ……

a_2 b_2 DATA$_2$: p2×(**b_2**+7)/8 bytes

……



Number of DATAn = 12 × (8+7)/8 = **12 Bytes**

2-7-2 DD (Downloaded font Delete)

Description

Delete downloaded font from memory

Syntax

DD*'font name'*

Parameters

'font name' : Indicate the Image in memory to be deleted.(A~Z)

- ♣ **This name is Case Sensitive.**
- ♣ **By using *, all images in memory will be deleted.**
- ♣ **You can show the downloaded font list by DI command.**

Example

DD'A'	// Delete downloaded font A
DD*	// Delete all downloaded fonts in memory

2-7-3 DI (Downloaded font Information)

Description

Print list of downloaded font

Syntax

DI

Example

DI

Result

```
Download Font Information
============================
Name        w     h      c      Size
============================
C           16    25     128    6400
G           12    24     224    10752

Free Memory                     179419
============================
```

♣ **w** : font width, **h** : font height, **c**: total number of characters

2-8 The Others

Commands not included in 1 to 7 categories.

1) @

Printer initialization.

2) PI

Print information of printer configuration.

3) CUT

Enable/Disable Cutting Action

4) ^cp

Check printer status and report 2bytes status data to host.

5) ^cu

Check printer status and report 1byte status data to host.

6) ^PI

Send various printer information to host.

7) ^MBZ

Power save mode setting.

8) ^MBP

Send power save mode setting to host.

2-8-1 @ (Initialize Printer)

Description

Initialize the printer

Syntax

@

2-8-2 PI (Printer Information)

Description

Print current printer setting

Syntax

PI

2-8-3 CUT (Auto-cutter Enable/Disable)

Description

Enable or Disable Auto-cut action after printing by 'P' command

Syntax

CUT*p1(,p2)*

Parameters

p1 : Cutting Action Enable/Disable

y : Enable cutter to act after printing is finished.

n : Disable cutter.

p2 : Cutting Period

♣ **Cutting Period means the number of pages between two cuttings.**

♣ **This command is not the cutting command itself but cutting enable/disable command.**

♣ ***Cutting is executed immediately after printing is finished by P command if the cutter option is enabled by this CUT command.***

♣ **Last page is always cut.**

Example – p1(Cutter Enable/Disable)

Cutting is executed after Printing is finished	Cutting is not executed after Printing is finished
T20… B130… BD… … **CUTy** P1	T20… B130… BD… … **CUTn** P1

Example – p2(Cutting Period)

```
CUTy          // Cut every page
CUTy,1        // Cut every page
CUTy,2        // Cut every 2 pages
CUTy,4        // Cut every 4 pages
```

2-8-4 ^cp (Check Printer Status and Report 2 bytes)

Description

Check printer status and report 2bytes status data to host

Syntax

^cp

Return Value

1. Format

<1st Byte> <2nd Byte>

2. Table

Return Values		Description	Hex
Byte	**bit**		
1st Byte	7	Paper Empty	0x80
	6	Cover Open	0x40
	5	Cutter jammed	0x20
	4	Thermal Head(TPH) overheat.	0x10
	3	Gap Detection Error(Auto-sensing failure)	0x08
	2	Ribbon End Error	0x04
	1	Not assigned	0x02
	0	Not assigned	0x01
2nd Byte	7	On building label to be printed in image buffer.	0x80
	6	On printing label in image buffer	0x40
	5	Issued label is paused in peeler unit.	0x20
	4	Not assigned	0x10
	3	Not assigned	0x08
	2	Not assigned	0x04
	1	Not assigned	0x02
	0	Not assigned	0x01

3. Examples

When Return Values are		Printer Status is
1st Byte	**2nd Byte**	
0x00	0x00	No Error. The printer is ready to build and print label.
0x80	0x00	No paper is installed in printer.
0x80	0x40	Paper roll out while printing label. When new paper roll is loaded, the printer will re-issue the last label.
0x60	0x40	While printing, cutter is jammed and cover is opened (by user).

2-8-5 ^cu (Check Printer Status and Report 1 byte)

Description

Check printer status and report 1 byte status data to host

Syntax

^cu

Return Value

1. Format

<1st Byte>

2. Table

Return Values		Description	Hex
Byte	bit		
1st Byte	7	Paper Empty	0x80
	6	Cover Open	0x40
	5	Cutter jammed	0x20
	4	Thermal Head(TPH) overheat.	0x10
	3	Gap Detection Error(Auto-sensing failure)	0x08
	2	Ribbon End	0x04
	1	Not assigned	0x02
	0	Not assigned	0x01

2-8-6 ^PI (Send Printer information to host)

Description

Send various printer information such as model name, firmware version, statistics data or so to host

Syntax

^PI*p1(,p2)(,p3)*

Parameters

p1 : items to be reported.

0 : Model Name

1 : Model Type : Disabled

2 : F/W Version

3: None

4 : Mechanical conditions of printer

p2	Item	Unit
0	TPH temperature	℃
1	Printing density (density)	-
2	Tear-off/cutter position	dot

Return Value Format

Items	Return Format	Example
Model Name	Character String + 0x0d + 0x0a	“SLP-D220” + 0x0d + 0x0a
Model Type	Disabled	
F/W Version	Character String + 0x0d + 0x0a	“1.23” + 0x0d + 0x0a
TPH temperature	Character String + 0x0d + 0x0a	“85” + 0x0d + 0x0a
Printing density (density)	Character String + 0x0d + 0x0a	“17” + 0x0d + 0x0a
Paper Width	Character String + 0x0d + 0x0a	“832” + 0x0d + 0x0a
Paper Length	Character String + 0x0d + 0x0a	“1200” + 0x0d + 0x0a
Gap Length	Character String + 0x0d + 0x0a	“24” + 0x0d + 0x0a
Paper Horizontal Margin	Character String + 0x0d + 0x0a	“10” + 0x0d + 0x0a
Paper vertical Margin	Character String + 0x0d + 0x0a	“12” + 0x0d + 0x0a
Tear-off/cutter position	Character String + 0x0d + 0x0a	“+80” + 0x0d + 0x0a

2-8-7 ^MBZ (Power Save mode setting)

Description

Power save mode setting

Syntax

^MBZ*p1,p2*

Parameters

p1 : Power save mode enable/disable

0: Disable

1: Enable

p2 : Power save mode penetration timely set (unit: seconds)

0 < p2 < 256

p2 sec.

Example

1) Power save mode enable, penetration timely is 30seconds.

^MBZ1,30

2) Power save mode disable

^MBZ0,0

2-8-8 ^MBP (Send power Save mode information to host)

Description

Send power save mode setting to host

Syntax

^MBP

Parameters

None

Return Value Format

“ Enable/disable” + “ penetration timely” + 0x0d + 0x0a

Example

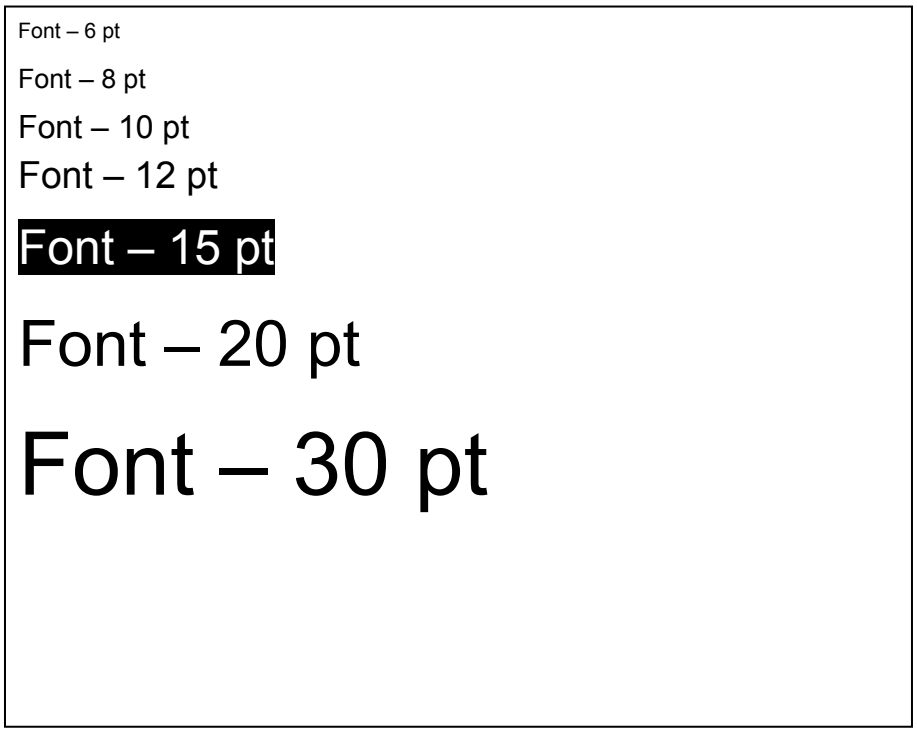
0x01 0x14 0x0d 0x0a // Power save mode enable (0x01)
// penetration timely 20seconds (0x14)

3. Programming Example

3-1 Example) T_resident

```
SS3                                          // Set Speed to 4 ips
SD20                                         // Set Density level to 20
SW800                                        // Set Label Width 800
SOT                                          // Set Printing Orientation from Top to Bottom
T26,20,0,1,1,0,0,N,N,'Font - 6 pt'
T26,49,1,1,1,0,0,N,N,'Font - 8 pt'
T26,81,2,1,1,0,0,N,N,'Font - 10 pt'
T26,117,3,1,1,0,0,N,N,'Font - 12 pt'
T26,156,4,1,1,0,0,R,N,'Font - 15 pt'
T26,200,5,1,1,0,0,N,N,'Font - 20 pt'
T26,252,6,1,1,0,0,N,N,'Font - 30 pt'
P1
```

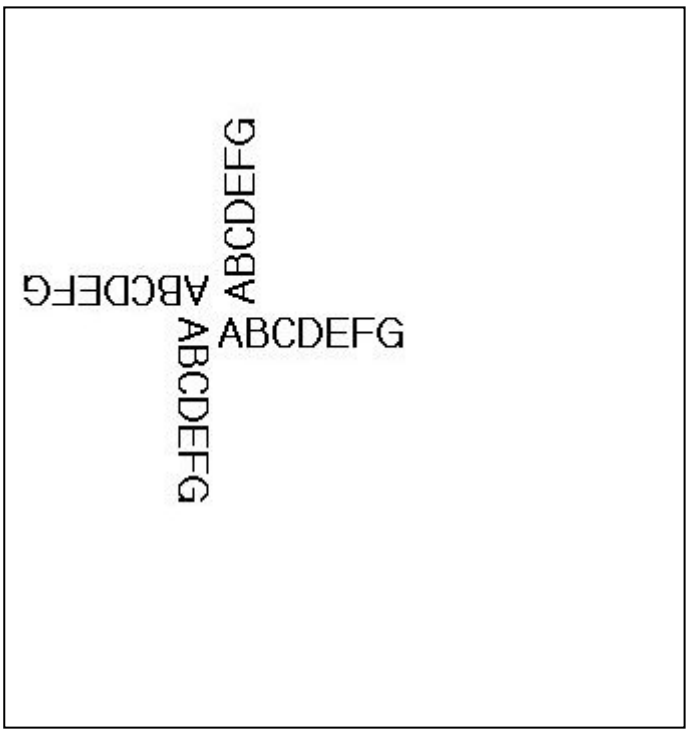
Result



3-2 Example) T_Rotate4

```
SS3
SW832
T300,500,4,1,1,0,0,N,N,'ABCDEFG'
T300,500,4,1,1,0,1,N,N,'ABCDEFG'
T300,500,4,1,1,0,2,N,N,'ABCDEFG'
T300,500,4,1,1,0,3,N,N,'ABCDEFG'
P1
```

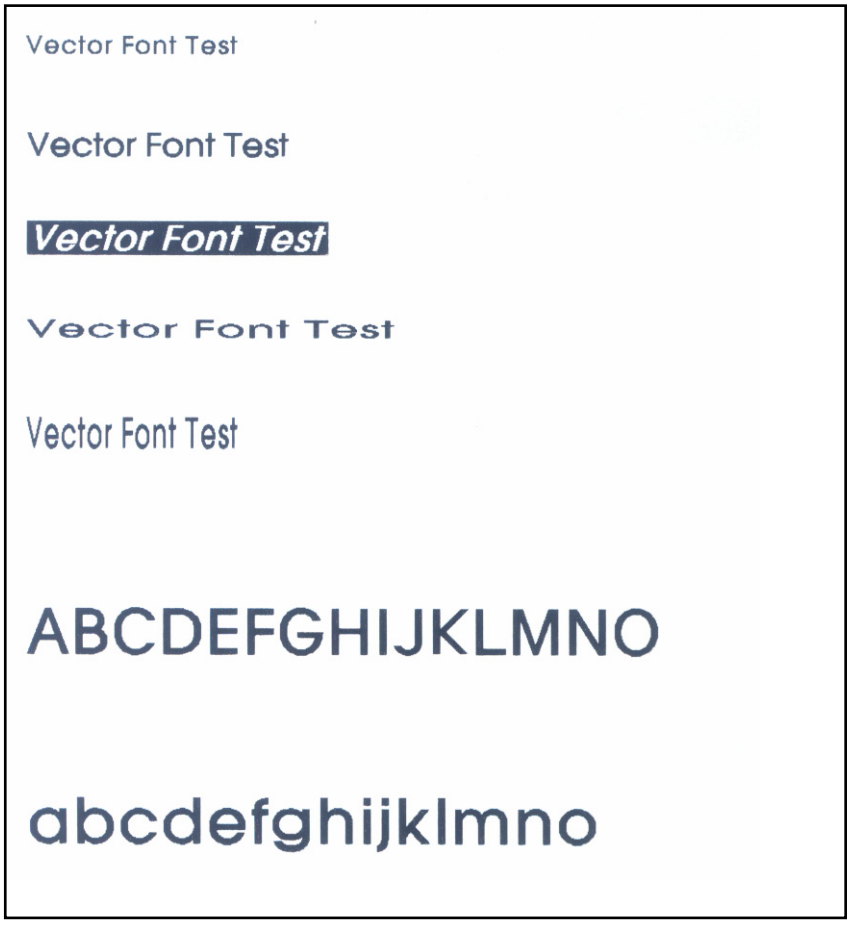
Result



3-3 Example) V_resident

```
SS3                                    // Set speed to 4 ips
SD20                                   // Set density to 20
SW800                                  // Set label width to 800
SOT                                    // Set printing direction to from top to bottom
V50,100,U,25,25,+1,N,N,N,0,L,0,'Vector Font Test'
V50,200,U,35,35,-1,N,N,N,0,L,0,'Vector Font Test'
V50,300,U,35,35,+1,B,R,I,0,L,0,'Vector Font Test '
V50,400,U,45,25,+1,N,N,N,0,L,0,'Vector Font Test'
V50,500,U,25,45,+1,N,N,N,0,L,0,'Vector Font Test'
V50,700,U,65,65,+1,N,N,N,0,L,0,'ABCDEFGHIJKLMNO'
V50,900,U,65,65,+1,N,N,N,0,L,0,'abcdefghijklmno'
P1
```

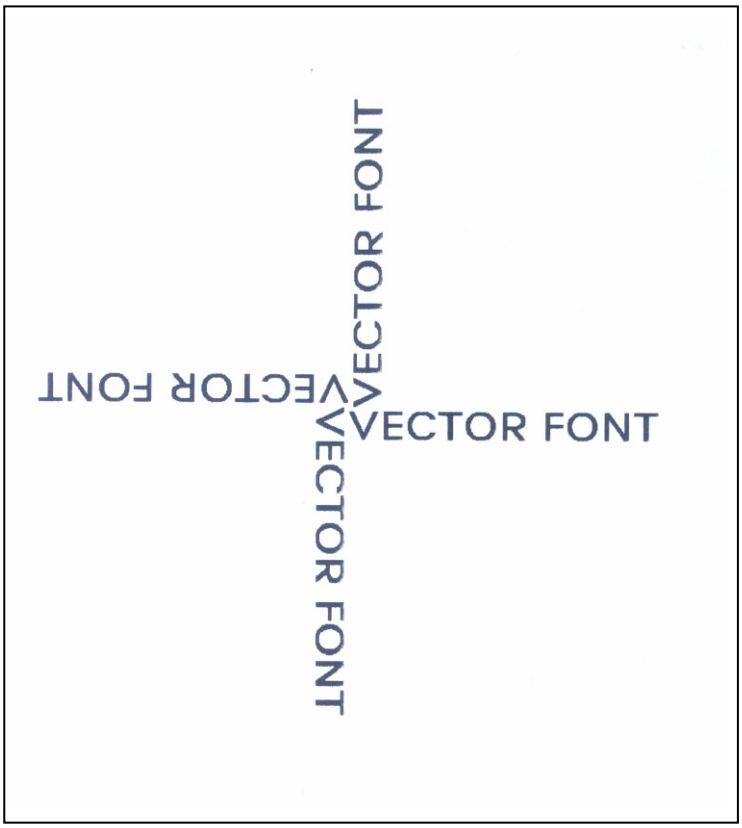
Result



3-4 Example) V_Rotate4

```
SS3
SW832
V400,500,U,45,40,+1,N,N,N,0,L,0,'VECTOR FONT'
V400,500,U,45,40,+1,N,N,N,1,L,0,'VECTOR FONT'
V400,500,U,45,40,+1,N,N,N,2,L,0,'VECTOR FONT'
V400,500,U,45,40,+1,N,N,N,3,L,0,'VECTOR FONT'
P1
```

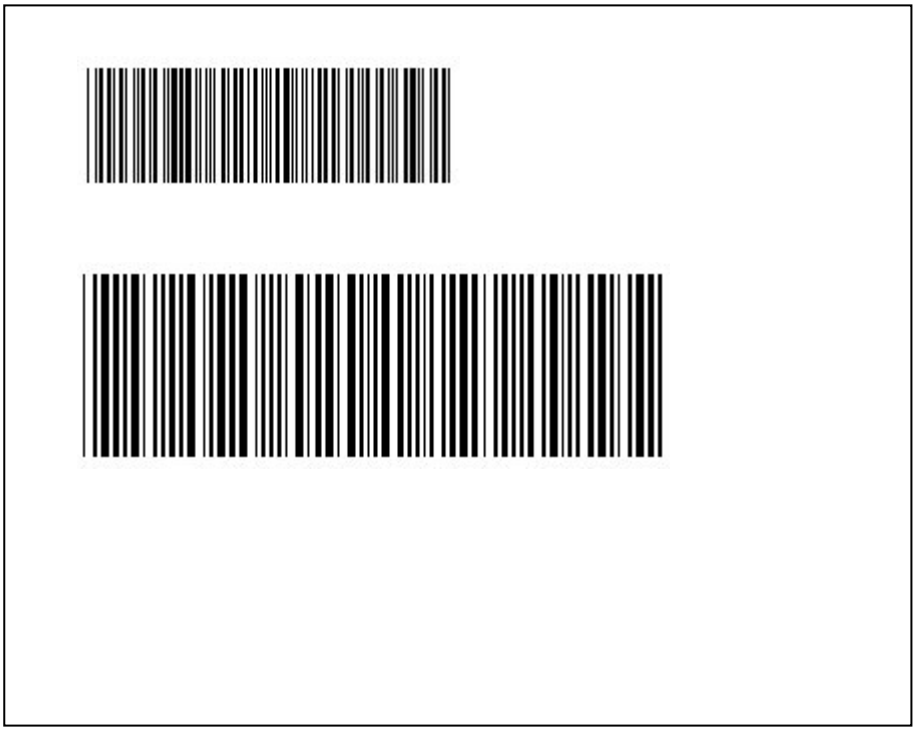
Result



3-5 Example) Code39

SM10,0
B178,196,0,2,6,100,0,0'1234567890' **// Caution : The position is not (178,196) but (78,196).**
B150,468,0,4,10,200,0,0'1234567890'
P1

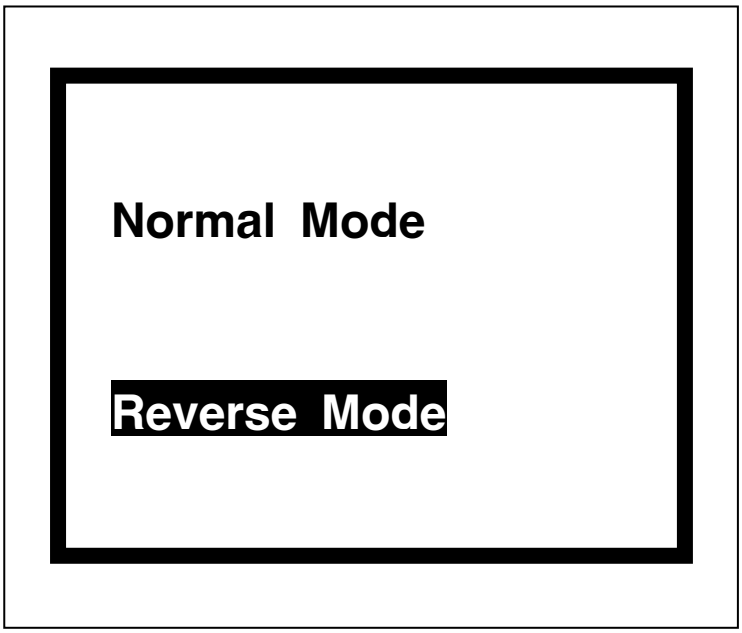
Result



3-6 Example) BD1

```
SS3                                  // Set Speed to 4 ips
SD20                                 // Set Density level to 20
SW800                                // Set Label Width to 800

BD50,50,750,500,B,20
T100,150,5,1,1,0,0,N,N,'Normal Mode'
T100,300,5,1,1,0,0,R,N,'Reverse Mode'

SOT
P1
```

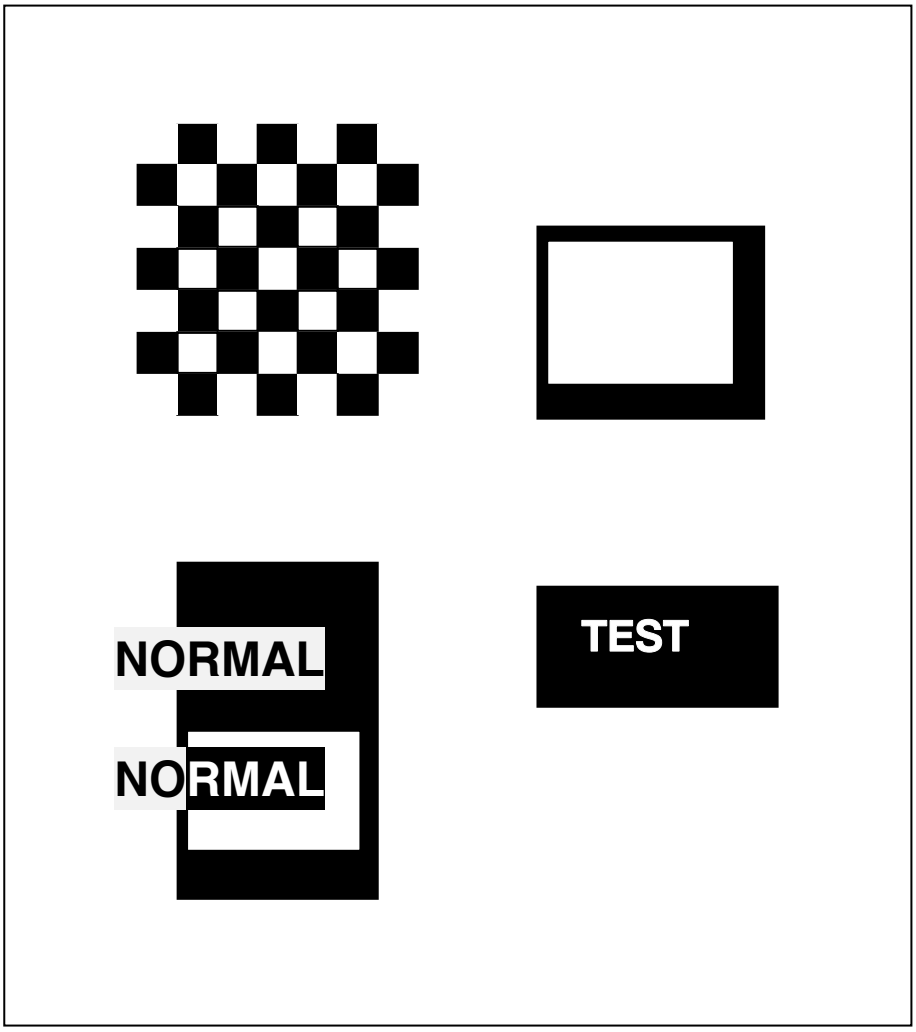
Result



3-7 Example) BD3

```
SS3                          // Set Printing Speed to 4 ips
SD20                         // Set Printing Density level to 20
SW800                        // Set Label Width to 800

BD50,100,400,150,O           // Draw a block in Overwriting Mode
BD50,200,400,250,O
BD50,300,400,350,O
BD100,50,150,400,E           // Draw a block in Exclusive OR mode
BD200,50,250,400,E
BD300,50,350,400,E
BD500,200,700,400,O
BD510,210,670,370,D          // Draw a block in Delete mode, namely Erase block
                                area
BD100,600,350,1000,O
T50,700,5,1,1,0,0,N,N,'NORMAL'   // Write Text data on image buffer
T50,800,5,1,1,0,0,N,N,'NORMAL'
BD110,780,340,900,E
T500,700,5,1,1,0,0,n,N,'TEST'
BD480,680,700,800,E

SOT                          // Set Printing Orientation from Top to Bottom
P1                           // Start Printing
```

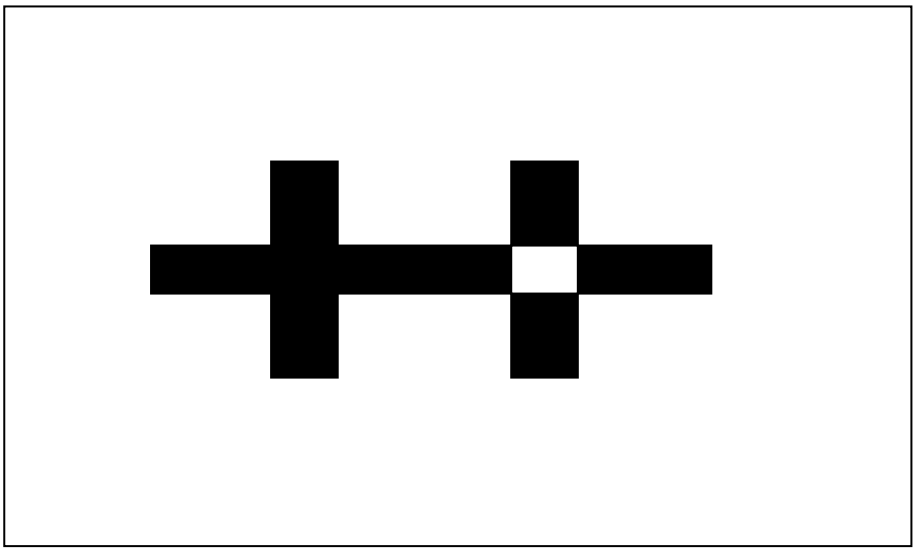
Result



3-8 Example) BD4

```
SW800
SM10,0

BD100,300,550,330,O          // Overwrite mode
BD200,200,250,430,O          // Overwrite mode
BD400,200,450,430,E          // Exclusive OR mode

P1
```

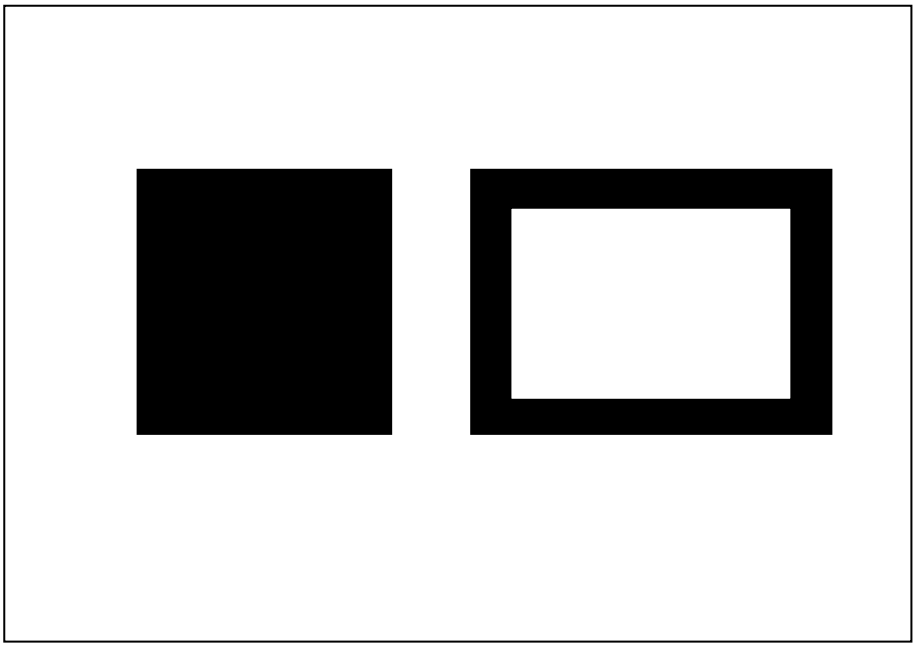
Result



3-9 Example) BD5

```
CB
SW800
SM10,0
BD100,300,300,500,O
BD400,300,700,500,B,30          // Box mode, additional parameter follows
P1
```

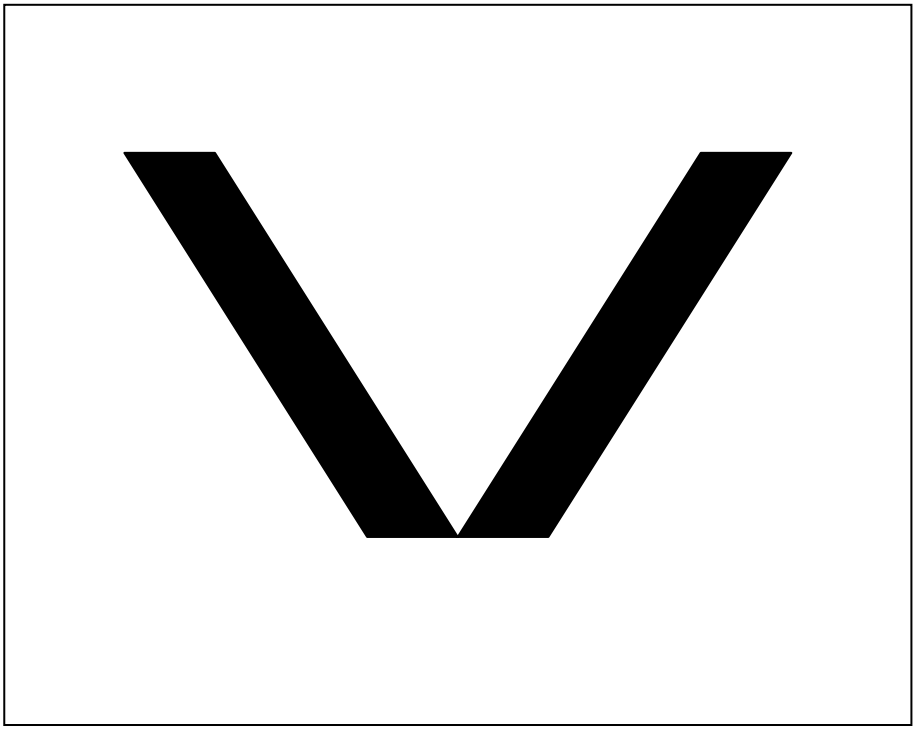
Result



3-10 Example) Slope

```
CB
SS3
SD20
SW8000

BD100,300,300,800,S,100          // Slope mode, additional parameter follows
BD600,300,400,800,S,100

P1
```

Result



3-11 Example) SW&SL

```
CB
SS3
SD20

SW800                                   // Set Label Width to 800
SL300,10,C                              // Continuous type
BD0,0,800,300,B,10
T30,40,4,1,1,0,0,N,N,'SW=800, SL=300, Continuous'
P1

SW600
SL500,10,C
BD0,0,600,500,B,10
T30,40,4,1,1,0,0,N,N,'SW=600, SL=500'
T30,100,4,1,1,0,0,N,N,'Continuous'
P1

SW400
SL800,10,C
BD0,0,400,800,B,10
T30,40,4,1,1,0,0,N,N,'SW=400'
T30,90,4,1,1,0,0,N,N,'SL=800'
T30,140,4,1,1,0,0,N,N,'Continuous'
P1
```

Result

SW = 800, SL=300, Continuous

SW = 600, SL=500,

Continuous

SW= 400,

SL = 800,

Continuous

3-12 Example) TEST00_TS

```
TD'Test00'                                  // Template Delete
TS' Test00'                                 // Start Template Store

SV00,15,N,'Manufacturer :'                  // Declare variable V00, field size:15, No justification
SV01,15,R,'Model Name :'                    // Declare variable V01, field size:15, Right justification
T50,100,3,1,1,0,0,N,N,'Manufacturer :'V00    // Print variable V00 with some text string
T50,150,3,1,1,0,0,N,N,'Model Name :'V01      // Print variable V01 with some text string
T50,300,3,1,1,0,0,N,N,V00                   // Print variable V00 only
T50,350,3,1,1,0,0,N,N,V01                   // Print variable V01 only

TE                                          // End Template Store

TI                                          // Print and show all templates in memory
```

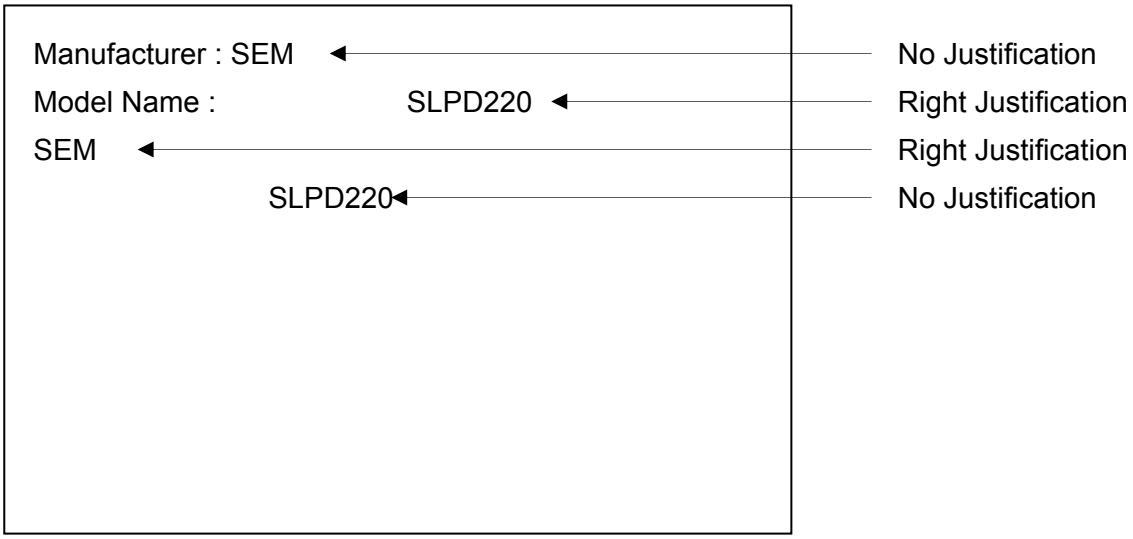
Result

```
Templates Information
====================
1. Test 1 <-------------------------- Stored in past time
2. Test0  <-------------------------- Stored in this time
Available template memory : 5.3Kbyte
```

3-13 Example) TEST00_TR

TR'**Test00'**	// Recall Stored template 'Test00'
?	// To get contents for variables used in 'Test00'
SEM	// Content for V00
SLPD220	// Content for V01
P1	// Print

Result

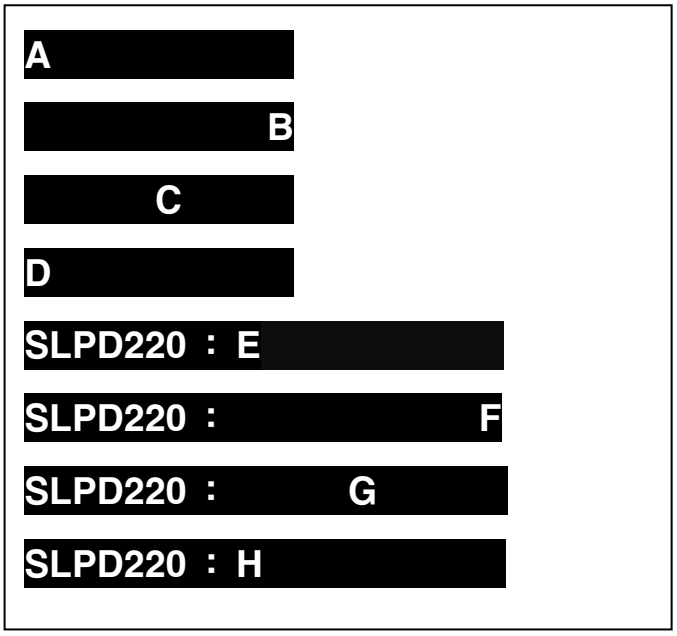


3-14 Example) TEST04_TS

```
TS'Test04'                          // Start Template Store

CB                                  // Clear Image Buffer
SS3                                 // Set Speed to 4 ips
SD20                                // Set Density level 20
SW800                               // Set Label Width to 800
SOT                                 // Set Printing Orientation from Top to Bottom(Default)

SV00,15,L,'prompt'                  // Declare variable V00, field size:15, Left justification
SV01,15,R,'prompt'                  // Declare variable V01, field size:15, Right justification
SV02,15,C,'prompt'                  // Declare variable V02, field size:15, Center
justification
SV03,15,N,'prompt'                  // Declare variable V03, field size:15, No justification
SV04,15,L,'prompt'                  // Declare variable V04, field size:15, Left justification
SV05,15,R,'prompt'                  // Declare variable V05, field size:15, Right justification
SV06,15,C,'prompt'                  // Declare variable V06, field size:15, Center
justification
SV07,15,N,'prompt'                  // Declare variable V07, field size:15, No justification

T26,50,4,1,1,0,0,R,N,V00            // Print variable only
T26,100,4,1,1,0,0,R,N,V01
T26,150,4,1,1,0,0,R,N,V02
T26,200,4,1,1,0,0,R,N,V03
T26,250,4,1,1,0,0,R,N,'SLPD220 :'V04    // Print variable with fixed text data
T26,300,4,1,1,0,0,R,N,' SLPD220 :'V05
T26,350,4,1,1,0,0,R,N,' SLPD220 :'V06
T26,400,4,1,1,0,0,R,N,' SLPD220 :'V07

TE                                  // End Template Store
```

3-15 Example) TEST04_TR

```
TR'Test04'      // Recall Template

?               // Start Get values for variables
A               // data for variable V00
B               // data for variable V00
C                    .
D                    .
E                    .
F                    .
G
H               // data for variable V07

P1              // Start Printing
```



3-16 Example) IR1

```
IR130,400,'BIXOLON'     // Recall stored image data
P1                      // Printing
```

!!! Use the PCXDown utility when you download the pcx image file to printer memory. Refer to IS command.

Result

3-17 Example) TEST10_TS

```
TS'Test10'                          // Start Template Store

CB                                  // Clear Image Buffer
SS3                                 // Set Speed to 5 ips
SD20                                // Set Density to 20
SW800                               // Set Label Width to 800
SOT                                 // Set Printing Orientation from Top to bottom

SV00,15,C,'prompt'                  // Declare Variable 00
SV01,15,N,'prompt'                  // Declare Variable 01
SV02,10,N,'prompt'                  // Declare Variable 02

T130,250,5,1,1,0,0,R,N,V00          // Print Content of V00
T250,600,5,1,1,0,0,N,N,V01          // Print Content of V01
IR130,400,V02                       // Use V02 as Image Name

TE                                  // End Template Store
```

3-18 Example) TEST10_TR

(File location : CD\Testfile\Template\Test10\TEST10_TR.txt)

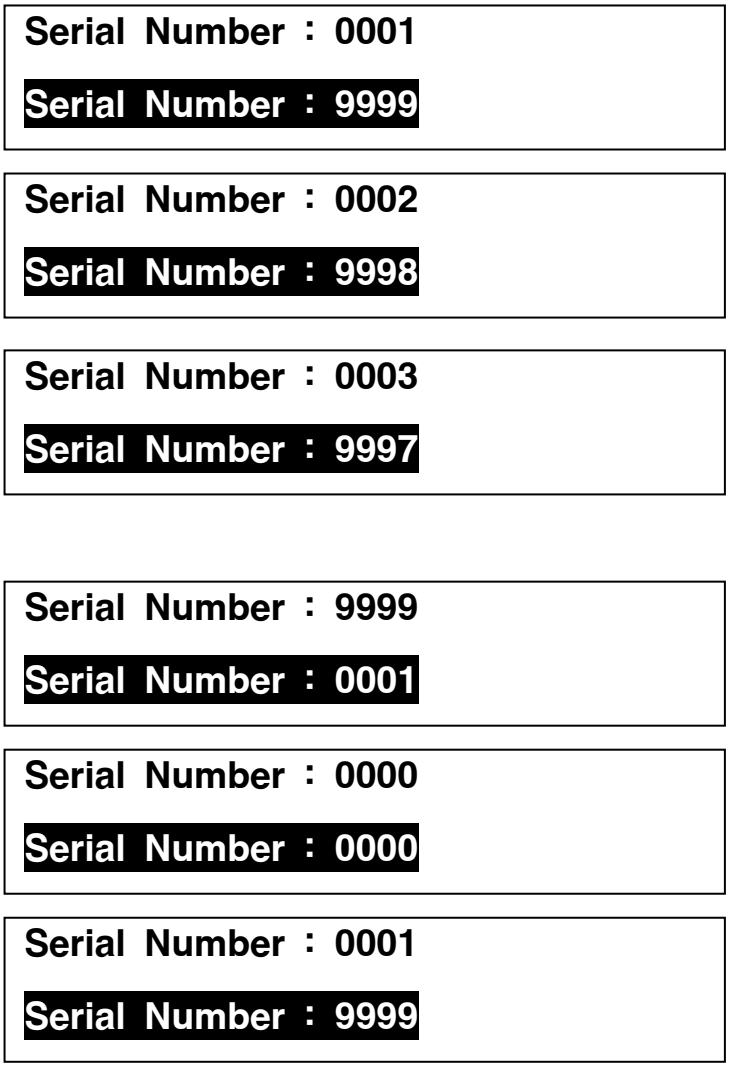
```
TR'Test10'      // Recall Template

?               // Start Get data for variables
BIXOLON         // data for V00
SLPD220         // data for V01
BIXOLON         // data for V02(Image Name)

P1              // Start Printing
```

3-19 Example) TEST11_TS

```
TS'Test11'                                  // Start Template Store

CB                                          // Clear Image Buffer
SS2                                         // Set Printing Speed to 4 ips
SD20                                        // Set Density to 20
SW800                                       // Set Label Width to 800
SOT                                         // Set Printing Orientation from Top to Bottom
SC0,4,L,+1,'COUNTER1'                       // Declare Counter 0, Field=4, Step:+1,Left
Justi.
SC1,4,N,-1,'COUNTER2'                       // Declare Counter 1, Field=4, Step: -1,No
Justi.
T50,50,4,1,1,0,0,N,N,'Serial Number : 'C0   // Print Counter 0
T50,150,4,1,1,0,0,R,N,'Serial Number : 'C1  // Print Counter 1

TE          // End of Template Store
```

3-20 Example) TEST11_TR

```
TR'Test11'      // Recall Template

?       // Start Get values for variables
0001    // data for Counter 0
9999    // data for Counter 1
P3,1

?
9999
0001
P3,1
```



3-21 Example) SLCS_BIXOLON

```
SM10,21
SS3
SD20
SW832
SOT
CS0,0

BD18,14,798,164,O
T400,62,4,2,2,0,0,R,B,'BIXOLON'
T65,98,3,1,1,0,0,R,B,'BIXOLON Label'
T20,276,3,1,1,1,0,N,N,'    BIXOLON'
T20,306,3,1,1,1,0,N,N,'    Yeongtong Dong'
T20,336,3,1,1,1,0,N,N,'    Sowon City,South Korea'
T22,218,4,1,1,0,0,N,B,'SHIP TO:'
BD18,410,784,415,O
BD553,197,558,413,O
B169,458,0,4,8,137,0,0,0,'*1234567890*'
T26,421,1,1,1,0,0,N,N,'POSTAL CODE:'
BD18,616,784,621,O
BD20,781,786,786,O
T503,798,1,1,1,0,0,N,N,'DESTINATION:'
T42,841,5,1,1,0,0,N,B,'30 Kg'
BD18,928,784,933,O
T25,798,1,1,1,0,0,N,N,'WEIGHT:'
T259,798,1,1,1,0,0,N,N,'DELIVERY NO:'
T23,630,1,1,1,0,0,N,N,'AWB:'
BD241,783,246,932,O
BD486,784,491,933,O
T274,841,5,1,1,0,0,N,B,'425518'
T104,627,3,1,1,0,0,N,N,'8741493121'
T565,841,5,1,1,0,0,N,B,'ICN'
B1127,672,4,4,8,90,0,0,0,'8741493121'
B2560,180,M,0,'999,840,06810,7317,THIS IS A TEST OF MODE 0 STRUCTURED CARRIER MESSAGE ENCODING. THIS IS AN 84 CHAR MSG'
B280,960,P,30,10,0,0,0,1,3,14,0,'BIXOLON Label Printer SLPD220, This is Test Printing.'
P1
```

Result


BIXOLON Label
BIXOLON
SHIP TO:
BIXOLON
Yeongtong Dong
Sowon City, South Korea
POSTAL CODE:
AWB: 8741493121
WEIGHT:
30 Kg
DELIVERY NO:
425518
DESTINATION:
ICN